



**ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ**  
**ΣΧΟΛΗ: ΜΗΧΑΝΙΚΩΝ ΗΛΕΚΤΡΟΝΙΚΩΝ**  
**ΥΠΟΛΟΓΙΣΤΩΝ, ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ ΚΑΙ**  
**ΔΙΚΤΥΩΝ**

**Διαλειτουργικότητα Των Πληροφοριακών**  
**Συστημάτων Στο Χώρο Της Υγείας: Το**  
**Πρωτόκολλο HL7**

**ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ**

Κλεονίκη Σ. Λύτρα

Βόλος, Οκτώβριος 2007



**ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ  
ΒΙΒΛΙΟΘΗΚΗ & ΚΕΝΤΡΟ ΠΛΗΡΟΦΟΡΗΣΗΣ  
ΕΙΔΙΚΗ ΣΥΛΛΟΓΗ «ΓΚΡΙΖΑ ΒΙΒΛΙΟΓΡΑΦΙΑ»**

Αριθ. Εισ.: 5997/1  
Ημερ. Εισ.: 01-11-2007  
Δωρεά: Συγγραφέα  
Ταξιθετικός Κωδικός: ΠΤ – ΜΗΥΤΔ  
2007  
ΛΥΤ



**ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ**  
**ΣΧΟΛΗ: ΜΗΧΑΝΙΚΩΝ ΗΛΕΚΤΡΟΝΙΚΩΝ**  
**ΥΠΟΛΟΓΙΣΤΩΝ, ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ ΚΑΙ**  
**ΔΙΚΤΥΩΝ**

**Διαλειτουργικότητα Των Πληροφοριακών**  
**Συστημάτων Στο Χώρο Της Υγείας: Το**  
**Πρωτόκολλο HL7**

**ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ**

**Κλεονίκη Σ. Λύτρα**

Επιβλέπων:

Εγκρίθηκε από εξεταστική επιτροπή την .....

.....

.....

.....

**Βόλος, Οκτώβριος 2007**

.....

Κλεονίκη Σ. Λύτρα

Αφιερωμένη στους γονείς μου, την αδερφή μου και τους φίλους μου.

Copyright © Κλεονίκη Σ. Λύτρα, 2007.

Με επιφύλαξη παντός δικαιώματος. All rights reserved.

Απαγορεύεται η αντιγραφή, αποθήκευση και διανομή της παρούσας εργασίας, εξ ολοκλήρου ή τμήματος αυτής, για εμπορικό σκοπό. Επιτρέπεται η ανατύπωση, αποθήκευση και διανομή για σκοπό μη κερδοσκοπικό, εκπαιδευτικής ή ερευνητικής φύσης, υπό την προϋπόθεση να αναφέρεται η πηγή προέλευσης και να διατηρείται το παρόν μήνυμα. Ερωτήματα που αφορούν τη χρήση της εργασίας για κερδοσκοπικό σκοπό πρέπει να απευθύνονται προς τον συγγραφέα.

Οι απόψεις και τα συμπεράσματα που περιέχονται σε αυτό το έγγραφο εκφράζουν τον συγγραφέα και δεν πρέπει να ερμηνευθεί ότι αντιπροσωπεύουν τις επίσημες θέσεις του Πανεπιστημίου Θεσσαλίας.



# **ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ**

## **ΠΕΡΙΛΗΨΗ**

## **ΕΙΣΑΓΩΓΗ- ΕΠΙΣΚΟΠΗΣΗ ΤΗΣ ΕΡΓΑΣΙΑΣ**

ΠΑΡΟΥΣΑ ΚΑΤΑΣΤΑΣΗ ΣΤΙΣ ΜΟΝΑΔΕΣ ΥΓΕΙΑΣ

ΤΕΧΝΟΛΟΓΙΕΣ ΕΠΙΚΟΙΝΩΝΙΑΣ ΤΩΝ ΙΑΤΡΙΚΩΝ

ΠΛΗΡΟΦΟΡΙΑΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

Η ΤΕΧΝΟΛΟΓΙΑ XML (eXtensible Markup Language)

ΤΟ ΠΡΟΤΥΠΟ HL7 Health Level 7

## **ΚΕΦΑΛΑΙΟ 1: ΙΑΤΡΙΚΑ ΠΛΗΡΟΦΟΡΙΑΚΑ ΣΥΣΤΗΜΑΤΑ**

1.1. ΕΙΣΑΓΩΓΗ

1.2. ΛΟΓΟΙ ΧΡΗΣΗΣ ΤΩΝ Ο.Π.Σ.Ν. (ΟΛΟΚΛΗΡΩΜΕΝΟ  
ΠΛΗΡΟΦΟΡΙΑΚΟ ΣΥΣΤΗΜΑ ΝΟΣΟΚΟΜΕΙΟΥ)

1.3. ΠΑΡΟΥΣΑ ΚΑΤΑΣΤΑΣΗ

1.4. ΠΑΡΑΓΟΝΤΕΣ ΑΝΑΠΤΥΞΗΣ

1.5. ΕΠΙΛΟΓΟΣ - ΣΥΜΠΕΡΑΣΜΑ

## **ΚΕΦΑΛΑΙΟ 2: ΔΙΑΛΕΙΤΟΥΡΓΙΚΟΤΗΤΑ**

2.1. ΟΡΙΣΜΟΣ

2.2. ΒΑΣΙΚΕΣ ΕΝΝΟΙΕΣ

2.3. ΥΠΗΡΕΣΙΑ ΛΟΓΙΣΜΙΚΟΥ

2.4. ΔΙΑΔΙΚΤΥΑΚΗ ΥΠΗΡΕΣΙΑ

2.4.1. ΟΡΙΣΜΟΣ

2.4.2. ΙΣΤΟΡΙΚΗ ΑΝΑΔΡΟΜΗ

2.4.3. ΜΕΙΟΝΕΚΤΗΜΑΤΑ MIDDLEWARE

#### 2.4.4. ΣΗΜΕΡΑ

### 2.5. ΤΟ ΠΡΩΤΟΚΟΛΛΟ SOAP (SIMPLE OBJECT ACCESS PROTOCOL)

#### 2.5.1. ΕΙΣΑΓΩΓΗ

#### 2.5.2. ΠΕΡΙΓΡΑΦΗ

#### 2.5.3. ΧΕΙΡΙΣΜΟΣ ΣΦΑΛΜΑΤΩΝ

#### 2.5.4. ΠΑΡΑΔΕΙΓΜΑ

#### 2.5.5. ΑΠΟΔΟΣΗ

### 2.6. WSDL (WEB SERVICES DESCRIPTION LANGUAGE)

#### 2.6.1. ΕΙΣΑΓΩΓΗ

#### 2.6.2. ΠΕΡΙΓΡΑΦΗ

#### 2.6.3. ΠΑΡΑΔΕΙΓΜΑ

### 2.7. UDDI (UNIVERSAL DESCRIPTION , DEFINITION AND INTEGRATION)

#### 2.7.1. ΕΙΣΑΓΩΓΗ

#### 2.7.2. ΠΕΡΙΓΡΑΦΗ

### 2.8. ΑΣΦΑΛΕΙΑ XML ΤΕΧΝΟΛΟΓΙΩΝ ΓΙΑ ΜΙΑ WEB SERVICE

## **ΚΕΦΑΛΑΙΟ 3: Η ΤΕΧΝΟΛΟΓΙΑ XML (EXTENSIBLE MARKUP LANGUAGE)**

### 3.1. ΟΡΙΣΜΟΣ - ΠΕΡΙΓΡΑΦΗ

### 3.3. ΧΑΡΑΚΤΗΡΙΣΤΙΚΑ

#### 3.3.1. ΕΤΙΚΕΤΕΣ

#### 3.3.2 DTD

#### 3.3.3. ΣΧΗΜΑ XML( XML Schema)

#### 3.3.4. ΟΡΘΟΤΗΤΑ XML ΕΓΓΡΑΦΟΥ

### 3.4. ΤΕΧΝΟΛΟΓΙΕΣ XML

### 3.5. XML ΚΑΙ HL7

#### 3.5.1. ΙΣΤΟΡΙΚΗ ΑΝΑΔΡΟΜΗ

#### 3.5.2. ΚΑΝΟΝΕΣ ΓΙΑ ΤΑ XML ΜΗΝΥΜΑΤΑ

#### 3.5.3. ΠΑΡΑΔΕΙΓΜΑ

3.5.4. XML ΜΕΤΑΣΧΗΜΑΤΙΣΜΟΙ

3.5.5. ΠΛΕΟΝΕΚΤΗΜΑΤΑ ΤΗΣ ΧΡΗΣΗΣ ΤΗΣ XML ΓΙΑ ΤΗΝ  
ΚΩΔΙΚΟΠΟΙΗΣΗ ΜΗΝΥΜΑΤΩΝ HL7

## **ΚΕΦΑΛΑΙΟ 4: ΤΟ ΠΡΟΤΥΠΟ HL7**

4.1. ΓΕΝΙΚΑ

4.2. ΤΙ ΣΗΜΑΙΝΕΙ Η ΟΝΟΜΑΣΙΑ HL7

4.3. ΓΕΝΙΚΗ ΠΕΡΙΓΡΑΦΗ ΤΟΥ ΠΡΩΤΟΚΟΛΛΟΥ

4.4. ΠΕΡΙΓΡΑΦΗ ΑΡΧΙΤΕΚΤΟΝΙΚΗΣ

4.4.1. ΕΙΣΑΓΩΓΗ

4.4.2.FIELD – ΠΕΔΙΑ

4.4.3. SEGMENT DEFINITION

4.4.4. MESSAGE DEFINITION

4.4.5. MESSAGE TYPE

4.4.6. FUNCTIONAL GROUP

4.4.7. ΠΑΡΑΔΕΙΓΜΑ ΜΗΝΥΜΑΤΟΣ HL7

## **ΚΕΦΑΛΑΙΟ 5: ΧΑΡΑΚΤΗΡΙΣΤΙΚΑ ΤΟΥ HL7**

5.1. ΕΙΣΑΓΩΓΗ

5.2. BACKGROUND

5.3. Η ΑΝΑΓΚΗ ΓΙΑ ΠΡΟΤΥΠΟ

5.4. ΣΤΟΧΟΙ ΤΟΥ ΠΡΟΤΥΠΟΥ

5.5. ΣΤΡΑΤΗΓΙΚΕΣ ΕΔΡΑΙΩΣΗΣ ΤΟΥ HL7

5.6. ΛΟΓΟΙ ΧΡΗΣΗΣ ΤΟΥ HL7

5.7. ΝΕΕΣ ΚΑΙ ΤΡΕΧΟΥΣΕΣ ΠΡΩΤΟΒΟΥΛΙΕΣ

5.7.1. HIPAA

5.7.2. ΑΝΑΦΟΡΙΚΟ ΜΟΝΤΕΛΟ ΠΛΗΡΟΦΟΡΙΩΝ – REFERENCE  
INFORMATION MODEL (RIM)

5.7.3. ΛΕΞΙΛΟΓΙΟ

5.7.4. XML



5.8. ΟΜΑΔΕΣ ΕΡΓΑΣΙΑΣ ΤΟΥ HL7

5.8.1. SIG-HIM : HEALTHCARE INFORMATION MANAGEMENT

5.8.2. SIG-TIG: TECHNICAL IMPLEMENTATION GUIDELINES

5.9. ΠΕΡΙΟΧΕΣ ΕΦΑΡΜΟΓΗΣ ΤΟΥ HL7

5.10. ΒΑΣΙΚΑ ΣΤΟΙΧΕΙΑ ΤΗΣ ΕΚΔΟΣΗΣ 3

5.10.1.CDA

5.10.2. MDF

5.10.3. RIM

5.10.4.CPRS

5.10.5. UPR

5.11. ΓΕΝΙΚΗ ΕΠΙΣΚΟΠΗΣΗ ΤΟΥ ΠΡΟΤΥΠΟΥ

5.11.1. ΕΙΣΑΓΩΓΗ

5.11.2. ΚΑΝΟΝΕΣ ΚΩΔΙΚΟΠΟΙΗΣΗΣ ΤΟΥ HL7

5.11.3. ΚΩΔΙΚΕΣ ΠΟΛΛΩΝ ΧΑΡΑΚΤΗΡΩΝ

5.11.4. ΑΝΤΙΜΕΤΩΠΙΣΗ ΔΙΑΦΟΡΟΠΟΙΗΣΕΩΝ

5.11.5. ΝΕΕΣ ΕΚΔΟΣΕΙΣ ΤΟΥ ΠΡΟΤΥΠΟΥ

5.11.6. ΜΕΤΑΦΟΡΑ ΑΡΧΕΙΩΝ ( file transfers )

5.11.7. ΣΥΝΕΡΓΑΣΙΑ ΜΕ ΑΛΛΑ ΠΡΩΤΟΚΟΛΛΑ

5.12. ΓΕΓΟΝΟΤΑ ΩΘΗΣΗΣ – ΣΚΑΝΔΑΛΗΣ (trigger events)

5.13. ACKNOWLEDGMENTS

5.14. ACKNOWLEDGMENTS : ENHANCED MODE

5.15. ΠΕΡΙΒΑΛΛΟΝΤΑ ΕΠΙΚΟΙΝΩΝΙΑΣ

5.16. ΚΑΝΟΝΕΣ ΔΗΜΙΟΥΡΓΙΑΣ ΤΩΝ HL7 ΜΗΝΥΜΑΤΩΝ

5.17. ΠΕΡΙΓΡΑΦΗ ΤΩΝ ΕΦΑΡΜΟΓΩΝ ΤΟΥ HL7

5.17.1. ΑΝΑΠΑΡΑΣΤΑΣΗ ΜΗΝΥΜΑΤΩΝ ΣΤΟ HL7

5.17.2. PATIENT ADMINISTRATION – ΔΙΑΧΕΙΡΙΣΗ ΑΣΘΕΝΩΝ

5.17.3. ORDER ENTRY – ΕΙΣΑΓΩΓΗ ΠΑΡΑΓΓΕΛΙΩΝ

5.17.4. FINANCIAL MANAGEMENT – ΟΙΚΟΝΟΜΙΚΗ ΔΙΑΧΕΙΡΙΣΗ

5.17.5. QUERY – ΑΙΤΗΣΕΙΣ

5.17.6. OBSERVATION REPORTING – ΑΝΑΦΟΡΑ ΠΑΡΑΤΗΡΗΣΕΩΝ

5.17.7. MEDICAL RECORDS / INFORMATION MANAGEMENT – ΔΙΑΧΕΙΡΙΣΗ ΑΡΧΕΙΩΝ ΙΑΤΡΙΚΩΝ ΠΛΗΡΟΦΟΡΙΩΝ

5.17.8. SCHEDULING – ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ

5.17.9. MASTER FILES – ΑΡΧΕΙΑ ΔΙΑΧΕΙΡΙΣΤΗ

5.17.10. PATIENT REFERRAL – ΠΑΡΑΠΕΜΠΤΙΚΑ ΑΣΘΕΝΟΥΣ

5.17.11. CLINICAL LABORATORY AUTOMATION –

ΑΥΤΟΜΑΤΙΣΜΟΣ ΚΛΙΝΙΚΟΥ ΕΡΓΑΣΤΗΡΙΟΥ

5.17.12. PERSONNEL MANAGEMENT – ΔΙΑΧΕΙΡΙΣΗ

ΠΡΟΣΩΠΙΚΟΥ

5.17.13. PATIENT CARE – ΠΕΡΙΘΑΛΨΗ ΑΣΘΕΝΟΥΣ

## **ΚΕΦΑΛΑΙΟ 6: ΕΞΕΛΙΞΗ ΤΟΥ ΠΡΟΤΥΠΟΥ**

HL7 VERSION 3

HL7 VERSION 2.5

HL7 VERSION 2.4

HL7 VERSION 2.3.1

HL7 VERSION 2.3

ΠΛΕΟΝΕΚΤΗΜΑΤΑ ΤΩΝ ΕΚΔΟΣΕΩΝ 2.x ΤΟΥ HL7

ΜΕΙΟΝΕΚΤΗΜΑΤΑ ΤΩΝ ΕΚΔΟΣΕΩΝ 2.x ΤΟΥ HL7

## **ΚΕΦΑΛΑΙΟ 7: ΠΕΙΡΑΜΑΤΙΚΗ ΕΦΑΡΜΟΓΗ**

7.1. ΧΑΡΑΚΤΗΡΙΣΤΙΚΑ

7.2. ΤΕΧΝΟΛΟΓΙΚΟ ΠΕΡΙΒΑΛΛΟΝ

7.3. ΠΕΡΙΒΑΛΛΟΝ ΛΕΙΤΟΥΡΓΙΑΣ

## **ΚΕΦΑΛΑΙΟ 8: ΕΠΙΛΟΓΟΣ**

8.1. ΣΥΜΠΕΡΑΣΜΑΤΑ

8.2. ΜΕΛΛΟΝΤΙΚΕΣ ΕΠΕΚΤΑΣΕΙΣ

## **ΚΕΦΑΛΑΙΟ 9: ΒΙΒΛΙΟΓΡΑΦΙΑ**

### **ΠΑΡΑΡΤΗΜΑ : ΚΩΔΙΚΑΣ**

## ΠΕΡΙΛΗΨΗ

Ο σκοπός της διπλωματικής αυτής εργασίας είναι η εκτεταμένη μελέτη και περιγραφή του προτύπου HL7, των τεχνολογιών που διευκολύνουν τη διαλειτουργικότητα των πληροφοριακών συστημάτων, η διερεύνηση της ανάπτυξης μιας υλοποίησης του προτύπου HL7 με XML τεχνολογίες. Ένας ακόμα στόχος της προσέγγισης αυτής είναι και η επισήμανση-καταγραφή των πλεονεκτημάτων και μειονεκτημάτων χρήσης του HL7 προτύπου (και στην Ελλάδα) καθώς και η παράθεση επιπλέον λειτουργιών και δυνατοτήτων που προκύπτουν από την υιοθέτηση XML τεχνολογιών για την ανάπτυξη εφαρμογών που είναι συμβατές με το HL7. Διερευνώνται και αναλύονται πολλές διαφορετικές XML τεχνολογίες, οι κυριότερες από τις οποίες είναι: η XML Schema Language (εξετάζεται ως προς τον τρόπο επικύρωσης των HL7 μηνυμάτων) και ο XSLT μετασχηματισμός (εξετάζεται ως προς τη μορφοποίηση των HL7 μηνυμάτων όταν είναι γραμμένα σε XML μορφή).

### Λέξεις Κλειδιά:

HL7, XML, XSLT, XSD, XML Schema transformations, XML technologies, Health Level Seven, SOAP, WSDL, XPath, Schematron, validation, parsing.



## ΕΙΣΑΓΩΓΗ - ΕΠΙΣΚΟΠΗΣΗ ΤΗΣ ΕΡΓΑΣΙΑΣ

Σκοπός της διπλωματικής αυτής εργασίας θα είναι η μελέτη της επικοινωνίας ανάμεσα σε απομακρυσμένα πληροφοριακά συστήματα στο χώρο της Υγείας – Πρόνοιας. Πιο συγκεκριμένα στόχοι μας θα είναι:

- Η μελέτη του προβλήματος της διαλειτουργικότητας των Ιατρικών Πληροφοριακών Συστημάτων.
- Η ανάλυση των λεγόμενων Διαδικτυακών Υπηρεσιών, μέσω των οποίων εδραιώνεται η επικοινωνία ετερογενών και απομακρυσμένων Πληροφοριακών Συστημάτων.
- Η παρουσίαση και η μελέτη της τεχνολογίας XML (eXtensible Markup Language) καθώς και της σχέσης που υπάρχει ανάμεσα στην τεχνολογία XML και στα πληροφοριακά συστήματα.
- Η παρουσίαση και ανάλυση του προτύπου HL7 (Health Level 7).
- Η μελέτη μιας εφαρμογής, μέσω της οποίας θα θέλαμε να προτείνουμε έναν τρόπο ανταλλαγής των HL7 μηνυμάτων.

Αρχικά θα εξετάσουμε την παρούσα κατάσταση στις μονάδες υγείας και τον τρόπο επεξεργασίας και αποστολής των ιατρικών δεδομένων των ασθενών. Στη συνέχεια, θα παρουσιάσουμε τρεις διαφορετικές τεχνολογίες επίτευξης της επικοινωνίας ανάμεσα σε απομακρυσμένα ιατρικά πληροφοριακά συστήματα. Έπειτα, θα μιλήσουμε για την XML τεχνολογία, θα παρουσιάσουμε ορισμένα χαρακτηριστικά της και θα δούμε τον τρόπο χρήσης της XML στο πρωτόκολλο HL7. Στην πορεία, θα παραθέσουμε αναλυτικά το πρωτόκολλο HL7 με όλα τα χαρακτηριστικά και τις σχετικές πληροφορίες με αυτό. Το επόμενο βήμα θα είναι η παρουσίαση μιας μικρής πειραματικής εφαρμογής, μέσω της οποίας θέλουμε να επιτύχουμε την ανταλλαγή ενός HL7 μηνύματος, μεταξύ ετερογενών ιατρικών πληροφοριακών συστημάτων, με τη χρήση της XML τεχνολογίας. Τέλος, θα

παραθέσουμε τα συμπεράσματά μας από την πειραματική μας εφαρμογή και τις πηγές της έρευνάς μας.

Ακολουθούν μερικές γενικές πληροφορίες σχετικά με τα πεδία που θα καλύψει η εργασία μας, χωρίς πολλούς τεχνικούς όρους, ώστε ο αναγνώστης μας να αποκτήσει μία πρώτη ιδέα.

## **ΠΑΡΟΥΣΑ ΚΑΤΑΣΤΑΣΗ ΣΤΙΣ ΜΟΝΑΔΕΣ ΥΓΕΙΑΣ**

*Ο* χώρος της πληροφορικής στον τομέα της υγείας- σε παγκόσμιο επίπεδο -είχε ανέκαθεν μια σειρά προβλημάτων τα οποία ήταν και είναι μοναδικά από την φύση τους. Στις περισσότερες περιπτώσεις ποτέ δεν υπήρχαν αρκετά χρήματα για επένδυση στην πληροφορική, όπως για παράδειγμα σε ολοκληρωμένα πληροφοριακά συστήματα νοσοκομείων. Η ανάγκη όμως για διαλειτουργικότητα, ανταλλαγή δεδομένων και συνεργασία αυτών των συστημάτων είναι επιτακτική σε αυτό το χώρο.

*Η* αντιμετώπιση αυτού του προβλήματος με τη συνεχή αντικατάσταση πληροφοριακών συστημάτων λύνει ίσως το τεχνολογικό πρόβλημα της μηχανοργάνωσης των υγειονομικών μονάδων αλλά δεν αποτελεί οριστική λύση η οποία να είναι συμφέρουσα τόσο για την Πολιτεία όσο και για την κοινότητα των χρηστών. Οι μικροί προϋπολογισμοί για την πληροφορική, η μεγάλη επιστημονική εξειδίκευση κάποιων πληροφοριακών συστημάτων που τα καθιστά αναντικατάστατα ή στη καλύτερη περίπτωση αντικαθίστανται δύσκολα (καθώς απαιτείται ισχυρή επένδυση σε χρόνο και χρήμα) και οι παράλληλες πολύπλοκες επιχειρησιακές διεργασίες που υφίστανται στις μονάδες υγείας, συνηγορούν στο παραπάνω επιχείρημα.

Στη σημερινή εποχή, η πίεση για αλλαγές και βελτιώσεις αυξάνεται ολοένα και περισσότερο. Το χάσμα ανάμεσα στη ζήτηση για ποιοτικές υπηρεσίες υγείας από τους πολίτες και στην ποιότητα της προσφοράς υπηρεσιών υγείας από την πολιτεία και τις μονάδες υγείας, ολοένα και μεγαλώνει. Έτσι, γίνεται πλέον επιτακτική η ανάγκη για ανταλλαγή και εύκολη πρόσβαση στα δεδομένα ενός ασθενούς, από απομακρυσμένα και ανεξάρτητα συστήματα. Η ανάγκη αυτή έκανε το χώρο της υγείας πρωτοπόρο σε παγκόσμιο επίπεδο στη δημιουργία προτύπων (standards) και στον τομέα της επιχειρησιακής ολοκλήρωσης εφαρμογών.

### **ΤΕΧΝΟΛΟΓΙΕΣ ΕΠΙΚΟΙΝΩΝΙΑΣ ΤΩΝ ΙΑΤΡΙΚΩΝ** **ΠΛΗΡΟΦΟΡΙΑΚΩΝ ΣΥΣΤΗΜΑΤΩΝ**

Αρχικά τα απομακρυσμένα ιατρικά πληροφοριακά συστήματα επικοινωνούσαν διαμέσου πρωτοκόλλων, τα οποία ανήκουν στην κατηγορία λογισμικού middleware. Τα πρωτόκολλα αυτά εμφανίστηκαν στις αρχές της δεκαετίας του '90. Τα λογισμικά αυτής της κατηγορίας χρησιμοποιούνται μέχρι και σήμερα αλλά παρουσιάζουν ένα πολύ σημαντικό μειονέκτημα. Οι εφαρμογές που υποστηρίζουν εξαρτώνται από τον εκάστοτε κατασκευαστή του λογισμικού.

Προκειμένου λοιπόν να ξεπεραστεί αυτό το πρόβλημα αναπτύχθηκαν οι Διαδικτυακές Υπηρεσίες. Οι Διαδικτυακές Υπηρεσίες (Web Services) είναι υπηρεσίες λογισμικού, οι οποίες αναγνωρίζονται από το URL τους και το περιβάλλον διεπαφής τους. Οι ενέργειές τους περιγράφονται σε XML (eXtensible Markup Language) και μπορούν να χρησιμοποιηθούν από πολλά διαφορετικά συστήματα. Τέλος, είναι ανεξάρτητες πλατφόρμας.



*Οι Διαδικτυακές Υπηρεσίες βασίζονται στις τεχνολογίες SOAP (Simple Object Access Protocol), WSDL (Web Services Description Language) και UDDI (Universal Description, Discovery and Integration).*

*Το πρωτόκολλο SOAP είναι ένα πρότυπο που επιτρέπει την επικοινωνία των Διαδικτυακών Υπηρεσιών με άλλες εφαρμογές. Αυτό επιτυγχάνεται με την αποστολή και λήψη κατάλληλων μηνυμάτων. Αναπτύχθηκε το 1999 και εδραίωσε την επικοινωνία εφαρμογών μέσω του Διαδικτύου με τρόπο ανεξάρτητο πλατφόρμας και γλώσσας προγραμματισμού της κάθε εφαρμογής.*

*Η WSDL αναπτύχθηκε από την IBM και την Microsoft και είναι μια πρότυπη γλώσσα που βασίζεται στην XML και περιγράφει τα δομικά στοιχεία που συνθέτουν μία Διαδικτυακή Υπηρεσία.*

*Το UDDI είναι ένα πρωτόκολλο καταχώρησης για Διαδικτυακές Υπηρεσίες. Προσφέρει την υποδομή για τη δημοσίευση και ανάκληση των Διαδικτυακών Υπηρεσιών με τη χρήση ενός συγκεκριμένου API. Κάθε καταχώρηση στο UDDI περιέχει το αρχείο wsdl της υπηρεσίας, τη διεύθυνση στην οποία βρίσκεται η υπηρεσία στο Διαδίκτυο και άλλες πληροφορίες σχετικές με τον κάτοχο της υπηρεσίας και την πολιτική χρήσης της.*

### **Η ΤΕΧΝΟΛΟΓΙΑ XML (eXtensible Markup Language)**

*Η XML αναπτύχθηκε από το W3C's XML Working Group το 1996. Είναι ένα βιομηχανικό πρότυπο, ένας τρόπος αναπαράστασης δεδομένων ανεξάρτητος από συγκεκριμένα συστήματα. Συνδυάζει την ισχύ και την επεκτασιμότητα της SGML (Standard Generalized Markup Language), από την οποία προέρχεται, με την απλότητα που απαιτεί το Διαδίκτυο. Όπως η HTML (HyperText Markup Language,*

γλώσσα σήμανσης υπερκειμένου), η XML εσωκλείει τα δεδομένα σε ετικέτες, αλλά υπάρχουν σημαντικές διαφορές μεταξύ των δύο γλωσσών. Δεν πρόκειται για μια γλώσσα προγραμματισμού αλλά για μία δηλωτική γλώσσα που ακολούθησε την ASN1. Μπορούμε να πούμε ότι η XML είναι ένα τυποποιημένο κείμενο για την αναπαράσταση δομημένης πληροφορίας στο Διαδίκτυο. Δεν είναι απαραίτητο, επομένως, να είναι κάποιος προγραμματιστής για να μπορέσει να την χρησιμοποιήσει ή να τη μάθει. Διευκολύνει τον υπολογιστή στην παραγωγή, ανάγνωση και στην εξασφάλιση της σαφήνειας των δεδομένων. Σε γενικές γραμμές, πρόκειται για ένα σύνολο κανόνων που ορίζουν το σχεδιασμό μορφών κειμένου για τη διευκόλυνση της δόμησης των δεδομένων.

### **ΤΟ ΠΡΟΤΥΠΟ HL7 (Health Level 7)**

Μπορούμε να πούμε ότι το HL7 πρότυπο είναι ένα εργαλείο, με ειδική εγγραφή, σε κάθε σύστημα, που επιτρέπει σε διαφορετικά συστήματα, ή ιατρικά μηχανήματα, να επικοινωνούν ακώλυτα μεταξύ τους, και να διαβιβάζουν τα δεδομένα που περιέχουν το ένα στο άλλο. Η επικοινωνία αυτή, τεχνικά ονομάζεται διαλειτουργικότητα. Το πρότυπο HL7 είναι λοιπόν ένας κώδικας επικοινωνίας, κοινά αποδεκτός, από όλους τους κατασκευαστές. Προσφέρει ουσιώδη βοήθεια στον ιατρό, αφού τον απαλλάσσει από τον περιττό φόρτο της “χειρωνακτικής” διαχείρισης τεράστιου όγκου ιατρικής πληροφορίας.

Το πρότυπο HL7 δεν αφορά αποκλειστικά στη διαβίβαση πληροφορίας, μεταξύ εργαστηρίου και κλινικής. Έχει δομηθεί έτσι, ώστε να συμπεριλαμβάνει, εκτός από κλινικά και εργαστηριακά δεδομένα, και κάθε άλλη πληροφορία σχετική με το εκάστοτε νοσηλευτικό ίδρυμα.

Το πρότυπο HL7 μπορεί να εγκατασταθεί και να λειτουργήσει στα ήδη υπάρχοντα πληροφοριακά συστήματα και στον ήδη υπάρχοντα ιατροτεχνολογικό εξοπλισμό. Δεν απαιτεί καμία αλλαγή, διασυνδέει τα συστήματα και τα μηχανήματα κάθε κατασκευαστή και για το λόγο αυτό είναι άμεσα και γενικά εφαρμόσιμο και στην Ελλάδα. Ό,τι είναι ήδη εγκατεστημένο σε ένα νοσοκομείο ή μια μονάδα υγείας, από πλευράς τεχνολογικού εξοπλισμού, κάθε είδους, με την χρήση του προτύπου HL7 συνδέεται και με τον ολόκληρο το υπόλοιπο εξοπλισμό. Αυτονόητο είναι, πως η εγκατάσταση του προτύπου HL7 στα νοσοκομεία και στους ασφαλιστικούς οργανισμούς, είναι έργο που αφορά στις αντίστοιχες υπηρεσίες πληροφορικής. Ο κλινικός και ο εργαστηριακός γιατρός δε χρειάζεται να υπεισέλθει καθόλου σε τεχνικά θέματα ή τεχνικές λεπτομέρειες.

## **ΠΕΙΡΑΜΑΤΙΚΗ ΕΦΑΡΜΟΓΗ**

Στο τέλος της εργασίας μας, θα προσπαθήσουμε να υλοποιήσουμε μία εφαρμογή, μέσω της οποίας ετερογενή και απομακρυσμένα ιατρικά πληροφοριακά συστήματα, θα μπορούν να επικοινωνήσουν και να ανταλλάξουν ιατρικά δεδομένα. Η επικοινωνία ανάμεσα στα δύο συστήματα θα γίνεται με την ανταλλαγή των κατάλληλων HL7 μηνυμάτων. Στη συνέχεια της εργασίας ο αναγνώστης θα διαπιστώσει ότι η δομή των HL7 μηνυμάτων είναι αρκετά περίπλοκη, γι αυτό το λόγο θα επιλέξουμε πριν γίνει η ανταλλαγή των ιατρικών δεδομένων, να τα μετατρέψουμε από την HL7 μορφή τους σε XML. Ο λόγος που θα επιλέξουμε την XML, όπως σε πολύ γενικές γραμμές αναφέραμε, είναι ότι πρόκειται για μία ευρέως χρησιμοποιούμενη τεχνολογία, η οποία αναγνωρίζεται από οποιαδήποτε πλατφόρμα.

*H* εφαρμογή μας θα δέχεται ως είσοδο το HL7 μήνυμα από το σύστημα στο οποίο ζητάται η πληροφορία, θα το μετατρέπει στην XML μορφή του και στη συνέχεια θα το παραδίδει με αυτή τη δομή στο καλών σύστημα.

# ΚΕΦΑΛΑΙΟ 1: ΙΑΤΡΙΚΑ ΠΛΗΡΟΦΟΡΙΑΚΑ ΣΥΣΤΗΜΑΤΑ

## 1.1. ΕΙΣΑΓΩΓΗ

*Η χρήση των Ιατρικών Πληροφοριακών Συστημάτων στην Ελλάδα μειονεκτεί συγκρινόμενη με τη χρήση τους σε άλλα προηγμένα κράτη. Στους χώρους της υγείας (νοσηλευτικά ιδρύματα, ασφαλιστικοί οργανισμοί) υπάρχουν εγκατεστημένες βάσεις δεδομένων διοικητικό-οικονομικών εφαρμογών κυρίως. Τα περισσότερα από αυτά τα συστήματα όμως είναι μάλλον «περασμένων τεχνολογιών» (text -based solutions, client server, mainframe) και δεν υποστηρίζουν ιατρικές εφαρμογές. Προγράμματα ιατρικών εφαρμογών έχουν εγκατασταθεί πιλοτικά, όμως παρουσιάζεται αντίσταση από τους ήδη υπάρχοντες χρήστες. Εν ολίγοις, έχουμε κυρίως νησίδες πληροφορικής (ακόμα και εντός μιας μονάδας υγείας).*

*Αναλυτικότερα, ένα πληροφοριακό σύστημα που είναι εγκατεστημένο και λειτουργεί προς όφελος ενός τομέα υγείας, ονομάζεται Ολοκληρωμένο Πληροφοριακό Σύστημα Νοσοκομείου (Ο.Π.Σ.Ν ).*

## 1.2. ΛΟΓΟΙ ΧΡΗΣΗΣ ΤΩΝ Ο.Π.Σ.Ν

*Οι λόγοι και τα οφέλη της εισαγωγής και χρήσης των πληροφοριακών συστημάτων στον τομέα της Υγείας είναι αρκετά προφανείς και ουσιώδεις. Αρχικά, μπορούμε να πούμε πως με τη χρησιμοποίηση των Ο.Π.Σ.Ν. παρατηρείται βελτίωση των παρεχόμενων υπηρεσιών (σε διοικητικό επίπεδο, διαγνωστικό*

επίπεδο και στο επίπεδο της διαχείρισης). Επιπλέον, προκύπτει άμεση μείωση της γραφειοκρατίας, αφού με την εισαγωγή των πληροφοριακών συστημάτων αυτόματα περιορίζονται οι χειρόγραφες διαδικασίες και κατά συνέπεια έχουμε τη βέλτιστη διαχείριση των καθημερινών ζητημάτων ενός νοσοκομείου. Τέλος, το κόστος περίθαλψης μειώνεται αφού γίνεται ορθολογικότερη διαχείριση των θεμάτων του κάθε ασθενούς και κατά συνέπεια αποφεύγονται οι άσκοπες ιατρικές πράξεις.

### **1.3. ΠΑΡΟΥΣΑ ΚΑΤΑΣΤΑΣΗ**

Σήμερα, στη χώρα μας έχει καταγραφεί ότι μεγάλο ποσοστό των νοσοκομείων και άλλων φορέων υγείας χρησιμοποιεί Διαχειριστικά Υποσυστήματα (Δ.Π.Σ.Ν.), ικανοποιητικός αριθμός παρόχων υγείας (της τάξης του 50 %) έχει εγκατεστημένα Πληροφοριακά Συστήματα Εργαστηρίων (ΠΣΕ) και ελάχιστος αριθμός κάνει χρήση των Ιατρικών Υποσυστημάτων (Ι.Π.Σ.Ν.).

Παρακάτω παρουσιάζουμε έναν πίνακα στον οποίο φαίνεται ο αριθμός των ιατρικών πληροφοριακών υποσυστημάτων σε 151 νοσοκομεία της Ελλάδας.

<b>Υποσυστήματα</b>	<b>Αριθμός Νοσοκομείων</b>	<b>Ποσοστό % επί του συνολικού</b>
ΔΠΣΝ	92	82%
ΙΠΣΝ	25	22%

ΠΣΕ	32	29%
Ακτινολογικά IS	2	2%

*Τέλος, αξίζει να αναφέρουμε ότι σε 118 νοσοκομεία στα οποία διεξήχθη σχετική έρευνα παρατηρήθηκε ότι 89 από αυτά δεν έχουν ιστοσελίδα (site), 23 διαθέτουν «πειραματικές» ιστοσελίδες, 6 διαθέτουν «επαγγελματικές» ιστοσελίδες, 43 νοσοκομεία δεν έχουν δυνατότητα χρήσης ηλεκτρονικού ταχυδρομείου (e-mail), 64 διαθέτουν μικρό αριθμό λογαριασμών e-mail (1 έως 20) και 4 (Πανεπιστημιακά) νοσοκομεία διαθέτουν πλήρη συστήματα υποστήριξης ηλεκτρονικού ταχυδρομείου.*

#### **1.4. ΠΑΡΑΓΟΝΤΕΣ ΑΝΑΠΤΥΞΗΣ**

*Η εισαγωγή και εδραίωση των πληροφοριακών συστημάτων στο χώρο της Υγείας επηρεάζεται από τους παρακάτω 6 κύριους παράγοντες:*

- Την οργανωτική δομή ενός συστήματος Υγείας: Ο τρόπος οργάνωσης ενός υγειονομικού φορέα σε όλα τα επίπεδα της λειτουργίας του (μηχανογράφηση, ύπαρξη-εκμετάλλευση ηλεκτρονικών υπολογιστών κλπ).
- Το θεσμικό και ρυθμιστικό πλαίσιο σχετικά με τη διαχείριση των ευαίσθητων-προσωπικών δεδομένων: Ο παράγοντας αυτός αφορά στην ύπαρξη του κατάλληλου θεσμικού πλαισίου για την εκμετάλλευση και ανταλλαγή των προσωπικών δεδομένων των ασθενών, εάν δεν υπάρχει η σωστή νομοθεσία, είναι σαφές ότι είναι αδύνατο να κοινοποιηθούν τα προσωπικά στοιχεία ενός ασθενούς, έστω και για ιατρικούς λόγους.

- Τη δυνατότητα προσφοράς σύγχρονων λύσεων από την Αγορά: Προκειμένου να εδραιωθούν τα υπολογιστικά συστήματα στους χώρους υγείας πρέπει η Αγορά, κάθε ενδιαφερόμενης χώρας, να είναι σε θέση όχι μόνο να προσφέρει αυτή τη δυνατότητα αλλά και να μπορεί να την υποστηρίξει. Είναι απαραίτητο δηλαδή να υπάρχει το κατάλληλα καταρτισμένο ανθρώπινο δυναμικό που θα είναι σε θέση να προμηθεύει τα κατάλληλα υπολογιστικά συστήματα και να τα υποστηρίζει όποτε χρειάζεται.
- Τη διορατικότητα και τη στρατηγική τη Πολιτείας στον τομέα της Υγείας-Πρόνοιας: Η εισαγωγή των Πληροφοριακών Συστημάτων στην Υγεία εξαρτάται σε πολύ μεγάλο βαθμό από το βαθμό εξοικείωσης, ενημέρωσης, διορατικότητας και συνειδητοποίησης της χρησιμότητάς τους από την πολιτεία και τους αρμόδιους φορείς.
- Τη φιλικότητα του περιβάλλοντος των εφαρμογών προς τον τελικό χρήστη: Τα εκάστοτε Πληροφοριακά Συστήματα θα πρέπει να προσφέρουν στον τελικό χρήστη ένα φιλικό και εύχρηστο περιβάλλον αλληλεπίδρασης.
- Τη Διασυνδεσιμότητα και Διαλειτουργικότητα των Εφαρμογών:
  - α)Πλαίσιο Διαλειτουργικότητας
  - β)HL7 (στην Υγεία Πρόνοια)
  - γ)Χρήση Κωδικοποιήσεων

## **1.5. ΕΠΙΛΟΓΟΣ – ΣΥΜΠΕΡΑΣΜΑ**

Τελειώνοντας το κεφάλαιο των ιατρικών πληροφοριακών συστημάτων είναι σημαντικό να πούμε ότι προκειμένου να επωφεληθούν οι φορείς της Υγείας & Πρόνοιας από αυτά πρέπει να ακολουθηθούν καθορισμένα βήματα. Για την επιτυχημένη και



αποτελεσματική χρήση των πληροφοριακών συστημάτων απαιτείται μακροχρόνιος σχεδιασμός, καθώς και η ύπαρξη ενός ισχυρού τμήματος πληροφορικής τόσο κατά τη διάρκεια ανάπτυξης τους όσο και κατά την περίοδο χρήσης τους στα νοσοκομεία. Επιπλέον, προκειμένου να εδραιωθούν στους τομείς της Υγείας & Πρόνοιας απαιτείται η συμμετοχή πολλών φορέων (πολιτεία και προσωπικό), η εκπαίδευση του προσωπικού, η τυποποίηση των δεδομένων αυτών των συστημάτων καθώς και η υψηλή ασφάλεια των δεδομένων αυτών. Τέλος, ευνόητο είναι πως καθοριστικό ρόλο στην ανάπτυξη των πληροφοριακών συστημάτων διαδραματίζει η συνεχής εισαγωγή και εκμετάλλευση των Νέων Τεχνολογιών.

## ΚΕΦΑΛΑΙΟ 2 : ΔΙΑΛΕΙΤΟΥΡΓΙΚΟΤΗΤΑ

### 2.1. ΟΡΙΣΜΟΣ

*Ο* όρος της *Διαλειτουργικότητας (Interoperability)* περιγράφεται όπως παρακάτω:

*Ι*κανότητα–δυνατότητα δύο ή περισσότερων συστημάτων να αλληλεπιδρούν, να ανταλλάσσουν πληροφορίες και να είναι σε θέση να τις χρησιμοποιούν.

### 2.2. ΒΑΣΙΚΕΣ ΕΝΝΟΙΕΣ

#### *Διασυνδεσιμότητα (Interfacing)*

*Δ*ιασύνδεση δύο ή περισσότερων συστημάτων προκειμένου να μεταφέρονται δεδομένα από το ένα στο άλλο.

#### *Ολοκλήρωση (Integration)*

*Δ*ιασύνδεση εφαρμογών με τρόπο διαφανή για το χρήστη. Οι εφαρμογές-υποσυστήματα ανταλλάσσουν δεδομένα ή χρησιμοποιούν κοινή βάση δεδομένων (ΒΔ).

Στη συνέχεια θα παρουσιάσουμε την έννοια της «Υπηρεσίας Λογισμικού» και της «Διαδικτυακής Υπηρεσίας», καθώς και κάποιων πρωτοκόλλων και τεχνολογιών, των οποίων η χρήση εξασφαλίζει τη διαλειτουργικότητα των πληροφοριακών συστημάτων.

### **2.3. ΥΠΗΡΕΣΙΑ ΛΟΓΙΣΜΙΚΟΥ**

*Η έννοια της «υπηρεσίας λογισμικού» (software service) τυπικά αφορά σε μία οντότητα λογισμικού η οποία προσφέρει ποικίλες λειτουργίες δεχόμενη αιτήσεις και αποστέλλοντας αντίστοιχες απαντήσεις-αποκρίσεις. Σύμφωνα με τον τυπικό αυτό ορισμό, μπορούμε να θεωρήσουμε ως υπηρεσίες λογισμικού ένα αντικείμενο στη γλώσσα προγραμματισμού Java ή μία αποθηκευμένη διαδικασία στην SQL. Συνήθως όμως με την έννοια υπηρεσία λογισμικού αναφερόμαστε σε προγράμματα-λογισμικά που προσφέρουν υπηρεσίες χαμηλού επιπέδου, οι οποίες χρησιμοποιούνται από άλλες εφαρμογές.*

### **2.4. ΔΙΑΔΙΚΤΥΑΚΗ ΥΠΗΡΕΣΙΑ**

#### **2.4.1.ΟΡΙΣΜΟΣ**

*Η έννοια «Διαδικτυακή Υπηρεσία» (Web Service) περιγράφει μία υπηρεσία λογισμικού που μπορεί να κληθεί από κάποια εφαρμογή, ανεξαρτήτως πλατφόρμας, λειτουργικού συστήματος και γλώσσας προγραμματισμού (βασική διαφορά τους από τις υπόλοιπες υπηρεσίες), διαμέσου των πρωτοκόλλων του Διαδικτύου (όπως το HTTP) και των XML τεχνολογιών.*

*Χρειάζεται ιδιαίτερη προσοχή όταν αναφερόμαστε στις Διαδικτυακές Υπηρεσίες και στους παρόχους υπηρεσιών λογισμικού (Application Service Providers: ASP). Υπάρχει μία ριζική διαφορά, η οποία έγκειται στο γεγονός ότι οι ASP παραδίδουν ολοκληρωμένες εφαρμογές από μία συγκεκριμένη κεντρική θέση φιλοξενίας, ενώ οι Web Services περιέχουν καταναμεμημένα στοιχεία εφαρμογών.*

Επιπλέον, οι ASP εφαρμογές εξαρτώνται από την υποδομή του μηχανήματος, κάτι που δε συμβαίνει στις Web Services.

#### **2.4.2. ΙΣΤΟΡΙΚΗ ΑΝΑΔΡΟΜΗ**

Οι πρώτες υπηρεσίες λογισμικού έκαναν την αρχική εμφάνιση τους μαζί με τις συμβολικές γλώσσες προγραμματισμού. Στη συνέχεια αναπτύχθηκαν γλώσσες υψηλότερου επιπέδου (Fortran) και το λειτουργικό σύστημα Unix με τη γλώσσα προγραμματισμού C και τα πρωτόκολλα TCP/IP. Η γλώσσα C έδωσε τη δυνατότητα δημιουργίας δικτυακών εφαρμογών για την επικοινωνία διεργασιών που εκτελούνταν σε διαφορετικά μηχανήματα. Στη συνέχεια εμφανίστηκαν οι γλώσσες λογικού προγραμματισμού και έπειτα ο αντικειμενοστραφής προγραμματισμός με τις γλώσσες C++ και Java.

Στις αρχές της δεκαετίας του '90 έκαναν την εμφάνιση τους πρωτόκολλα για την επικοινωνία κατανεμημένων δικτυακών συστημάτων. Τα πρωτόκολλα αυτά ανήκουν στην κατηγορία λογισμικού middleware. Το λογισμικό αυτού του τύπου αναφέρεται σε μία σειρά υπηρεσιών που επιτρέπουν την επικοινωνία διεργασιών που εκτελούνται σε διαφορετικές πλατφόρμες. Τα συστήματα με χρήση middleware χρησιμοποιούνται μέχρι και σήμερα και είναι πολύ χρήσιμα σε περιβάλλοντα στα οποία συνυπάρχουν ετερογενείς πλατφόρμες.

#### **2.4.3. ΜΕΙΟΝΕΚΤΗΜΑΤΑ MIDDLEWARE**

Η middleware αρχιτεκτονική παρουσιάζει αρκετά πλεονεκτήματα όσον αφορά στην επίτευξη της διαλειτουργικότητας μεταξύ διαφορετικών μηχανημάτων, όμως προκύπτουν και αρκετά μειονεκτήματα, τα πιο σημαντικά είναι:

- Ευρέως χρησιμοποιούμενες υπηρεσίες και συστήματα middleware χρησιμοποιούν τμήματα ιδιόκτητου λογισμικού, με συνέπεια πολλές εφαρμογές να εξαρτώνται από ένα συγκεκριμένο κατασκευαστή.
- Τα πρωτόκολλα COBRA και DCOM (πρωτόκολλα για χρήση middleware) δεν είναι συμβατά μεταξύ τους, συνεπώς απαιτείται επιπλέον διασύνδεση ανάμεσα στα δύο διαφορετικά middleware συστήματα.

#### **2.4.4. ΠΑΡΟΥΣΑ ΚΑΤΑΣΤΑΣΗ**

*Προκειμένου να ξεπεραστούν οι περιορισμοί που θέτουν οι τεχνολογίες middleware, έπρεπε να βρεθεί τρόπος να επιτρέπεται η από κοινού χρήση δεδομένων και εφαρμογών μέσα σε ένα δίκτυο. Έτσι αναπτύχθηκαν περισσότερο διαλειτουργικά πρότυπα. Ανάμεσα σε αυτά τα πρότυπα συγκαταλέγεται και η γλώσσα σήμανσης Extensible Markup Language (XML). Για τη συνεργασία και επικοινωνία των διαμοιραζόμενων εφαρμογών αναπτύχθηκαν οι Διαδικτυακές Υπηρεσίες.*

*Οι Διαδικτυακές Υπηρεσίες βασίζονται στις τεχνολογίες SOAP (Simple Object Access Protocol), WSDL (Web Services Description Language) και UDDI (Universal Description, Discovery and Integration).*

*Μία Διαδικτυακή Υπηρεσία (Web Service), όπως προαναφέραμε είναι μία υπηρεσία λογισμικού η οποία αναγνωρίζεται από το URL και το περιβάλλον διεπαφής της. Οι ενέργειές της περιγράφονται σε XML και η ίδια διαδικτυακή υπηρεσία μπορεί έπειτα να χρησιμοποιηθεί από άλλα συστήματα. Τα συστήματα αυτά τότε*

επικοινωνούν με την Web Service μέσω των τεχνολογιών του Διαδικτύου και χρησιμοποιούν XML τεχνολογίες.

Υπάρχουν αρκετοί ορισμοί για τις Διαδικτυακές Υπηρεσίες, όμως όλοι τους δίνουν τρία κοινά χαρακτηριστικά σε αυτές:

- Προσφέρουν στο χρήστη πολλές λειτουργίες και εφαρμογές μέσω των Web Services πρωτοκόλλων.
- Παρέχουν έναν αναλυτικό τρόπο περιγραφής των διεπαφών τους, ώστε ένας χρήστης να μπορέσει να αναπτύξει μία εφαρμογή τύπου client, προκειμένου να συνομιλήσει με μία Διαδικτυακή Υπηρεσία. Αυτή η περιγραφή συνήθως υπάρχει σε ένα XML κείμενο, το οποίο ονομάζεται Web Service Description Language (WSDL) κείμενο.
- Είναι καταχωρημένες, έτσι ώστε οι χρήστες να μπορούν να τις βρουν εύκολα. Αυτό επιτυγχάνεται με το Universal Discovery Description and Integration (UDDI).

Μερικές βασικές αρχές σχεδιασμού μίας Διαδικτυακής Υπηρεσίας είναι:

**Συναρμολoγησιμότητα:** Ένας προγραμματιστής μπορεί να διαλέξει ποια πρότυπα επιθυμεί χρησιμοποιήσει, ανάλογα με τις ανάγκες του. Όλες οι προδιαγραφές που περιλαμβάνονται σε μία Διαδικτυακή Υπηρεσία είναι απλές, εύχρηστες και μπορούν να συνδυαστούν προκειμένου να παρέχουν επιπλέον λειτουργικότητα.

**Γενική Λειτουργία:** Οι Διαδικτυακές Υπηρεσίες έχουν αναπτυχθεί σε XML, κάτι που μειώνει τον αριθμό των απαιτήσεων των εφαρμογών που μπορούν να υλοποιηθούν, είτε αυτές αναπτύσσονται

σε κοινούς server είτε σε έναν επιτραπέζιο υπολογιστή είτε σε πολύπλοκα ιατρικά συστήματα. Επιπλέον, υποστηρίζουν τόσο ασύγχρονες όσο και σύγχρονες επικοινωνίες.

**Βασίζονται σε ανεξάρτητα πρότυπα:** Ένας πολύ σημαντικός παράγοντας που βοηθά στη ραγδαία εξάπλωση και υιοθέτηση των προτύπων των Διαδικτυακών Υπηρεσιών, είναι το γεγονός πως αυτά τα πρότυπα βασίζονται σε άλλα όπως, το XML πρότυπο, το HTTP και το SOAP.

Στη συνέχεια θα παρουσιάσουμε εν συντομία τις XML τεχνολογίες στις οποίες βασίζεται μία Web Service:

## **2.5. ΠΡΩΤΟΚΟΛΛΟ SOAP**

### **2.5.1. ΕΙΣΑΓΩΓΗ**

Το πρωτόκολλο SOAP (Simple Object Access Protocol) είναι το νέο πρότυπο που επιτρέπει την επικοινωνία των Διαδικτυακών Υπηρεσιών με άλλες εφαρμογές. Αυτό επιτυγχάνεται με την αποστολή και λήψη κατάλληλων μηνυμάτων. Αναπτύχθηκε το 1999 προκειμένου να εδραιωθεί η επικοινωνία εφαρμογών μέσω του Διαδικτύου με τρόπο ανεξάρτητο πλατφόρμας και γλώσσας προγραμματισμού της κάθε εφαρμογής.

Μπορούμε να πούμε εν ολίγοις ότι πρόκειται για ένα “stateless” και μονόδρομο πρωτόκολλο επικοινωνίας, το οποίο αγνοεί τη σημασιολογία των μηνυμάτων που μεταφέρει και σχεδιάστηκε για να υποστηρίξει εφαρμογές που αλληλεπιδρούν ανταλλάσσοντας ασύγχρονα μηνύματα.

*Εν* ολίγοις, τα μηνύματά του περιγράφουν προγραμματιστικές ενέργειες για την επικοινωνία Διαδικτυακών Υπηρεσιών και εφαρμογών.

*Το* SOAP καθορίζει:

- Τη μορφή του μηνύματος για τη μονόδρομη επικοινωνία.
- Το σύνολο των κανόνων για τη χρήση SOAP μηνυμάτων. Οι κανόνες αυτοί προσδιορίζουν τον τρόπο κλήσης μιας απομονωμένης διαδικασίας με την αποστολή ενός SOAP μηνύματος καθώς και τον τρόπο απόκρισης της καλούμενης διαδικασίας μέσω ενός SOAP μηνύματος.
- Το σύνολο των κανόνων κωδικοποίησης SOAP μηνυμάτων. Εδώ καθορίζονται τα XML στοιχεία που πρέπει να διαβάσει και να κατανοήσει μία οντότητα καθώς και οι ενέργειες στις οποίες πρέπει να προβεί εάν δεν κατανοήσει το XML περιεχόμενο των μηνυμάτων.
- Τον τρόπο μεταφοράς του μηνύματος πάνω από το πρωτόκολλο μεταφοράς (HTTP ή SMTP).

*Για* να επιτευχθούν τα παραπάνω έχει αναπτυχθεί ένα μοντέλο σύμφωνα με το οποίο γίνεται η ανταλλαγή μηνυμάτων. Το μοντέλο αυτό θεωρεί ένα κατανεμημένο δικτυακό περιβάλλον που αποτελείται από ανεξάρτητους κόμβους. Οι κόμβοι αυτοί μπορεί να είναι είτε εφαρμογές είτε διεργασίες είτε Διαδικτυακές Υπηρεσίες που ανταλλάσσουν μεταξύ τους μηνύματα SOAP.

### **2.5.2. ΠΕΡΙΓΡΑΦΗ**

*Ένα* SOAP μήνυμα αποτελείται από:



- Μία επικεφαλίδα `http:` Περιλαμβάνει πληροφορίες δρομολόγησης για το μήνυμα και για τις τυχόν συναλλαγές που λαμβάνουν χώρα σε κάθε κόμβο. Πιο συγκεκριμένα υπάρχει το πεδίο `SOAPAction` που λαμβάνει ως τιμή το όνομα της μεθόδου της Web Service που καλείται και ελέγχεται από τον κόμβο που λαμβάνει το μήνυμα. Ακολουθεί παράδειγμα της επικεφαλίδας:

*POST /myserver HTTP/1.1*

*Host: www.mydomain.com*

*Content-Type: text/xml*

*Content-length:nnnn*

*SOAPAction: my-name-space#myMethod*

- Ένα έγγραφο XML με πρώτο στοιχείο το στοιχείο `<envelope>`. Στο στοιχείο αυτό περιέχεται ένα προαιρετικό υποστοιχείο, η επικεφαλίδα (header) και ένα υποχρεωτικό υποστοιχείο, το σώμα (body). Στο body περιέχονται τα δεδομένα για τις εντολές που πρέπει να εκτελεσθούν από την εφαρμογή που θα λάβει το μήνυμα. Τέλος, στο XML έγγραφο περιλαμβάνεται δήλωση για το χώρο των ονομάτων και το XML Schema που χρησιμοποιείται για τη μορφοποίηση του κειμένου.

Το πρωτόκολλο SOAP περιλαμβάνει ένα σύνολο κανόνων για τον τρόπο κωδικοποίησης οποιουδήποτε τύπου δεδομένων που μπορεί να αποσταλλεί σε μία εφαρμογή. Επίσης υποστηρίζει τόσο απλούς όσο και σύνθετους τύπους δεδομένων. Τα μηνύματα SOAP υλοποιούν κλήσεις RPC (Remote Procedure Call). Για την κλήση μιας μεθόδου ενός απομακρυσμένου αντικειμένου μιας Web Service στέλνεται ένα SOAP μήνυμα, που περιλαμβάνει στο σώμα του το όνομα της μεθόδου, τις παραμέτρους που πρέπει να δεχτεί και το URL αυτής. Μετά το πέρας της εκτέλεσης από την απομακρυσμένη Υπηρεσία, τα

αποτελέσματα επιστρέφονται στο καλών πρόγραμμα μέσω ενός άλλου SOAP μηνύματος.

*Η* επικοινωνία πραγματοποιείται ανάμεσα στον SOAP client (πελάτη) και στον SOAP server (διακομιστή). Ο πελάτης και ο εξυπηρετητής αναλαμβάνουν τη μετατροπή της μορφής του αιτήματος από τη μορφή που είναι κατανοητή σε μία πλατφόρμα, σε μορφή SOAP XML envelope και το αντίστροφο, δηλαδή λειτουργούν ως μεσάζοντες ανάμεσα στις δύο επικοινωνούντες μηχανές.

*Μία* επίσης χρήσιμη λειτουργία του SOAP server είναι η παραγωγή του αρχείου περιγραφής της Διαδικτυακής Υπηρεσίας (WSDL), το οποίο αποθηκεύει σε ειδικό μητρώο στο Διαδίκτυο (UDDI Registry). Όταν λοιπόν μία εφαρμογή client θέλει να καλέσει μία Web Service, αναζητά στο UDDI Registry την υπηρεσία που επιθυμεί και βάσει αυτού διαμορφώνει το κατάλληλο SOAP μήνυμα.

*Για* την μετάδοση των SOAP μηνυμάτων μπορεί να χρησιμοποιηθούν πολλά πρωτόκολλα, όπως το SMTP, το HTTPs και το JMS (Java Messaging Service), αρκεί αυτά τα πρωτόκολλα να υποστηρίζονται και από την καλούσα και από την καλούμενη εφαρμογή. Το πιο ευρέως χρησιμοποιούμενο πρωτόκολλο στην πράξη είναι το HTTP. Ο λόγος χρήσης του HTTP είναι η διαδεδομένη χρήση του στο Διαδίκτυο. Τα δεδομένα μεταφέρονται ακριβώς όπως οι ιστοσελίδες και δεν εμποδίζονται από προστασία ασφαλείας (π.χ. firewalls) που ενδέχεται να είναι εγκατεστημένα σε έναν υπολογιστή.

*Το* SOAP μήνυμα αποστέλλεται σαν μέρος του μηνύματος αίτησης ή απάντησης του HTTP (HTTP request και HTTP response, αντίστοιχα). Ο τρόπος αποστολής είναι ο HTTP-POST του HTTP, στην πορεία προστίθενται στο μήνυμα οι επικεφαλίδες HTTP (HTTP headers).

Μία RPC επικοινωνία αποτελείται από ζεύγη μηνυμάτων SOAP: το POST Request και το POST Response. Σε ένα POST Response μήνυμα περιέχεται το tag: <Result>, το οποίο παίρνει σαν τιμή το αποτέλεσμα της κλήσης μεθόδου και είναι αλφαριθμητικό (string).

Εκτός από την RPC λειτουργία ενός SOAP μηνύματος υπάρχει και μία δεύτερη. Το SOAP μήνυμα μπορεί να σταλεί και ως ειδοποίηση ή ενημέρωση για κάποια τρέχουσα κατάσταση από έναν κόμβο σε κάποιον άλλον, χωρίς να είναι απαραίτητο να σταλεί απάντηση. Τέλος, να σημειώσουμε ότι το SOAP είναι μια ειδική υλοποίηση του πρωτοκόλλου XML – RPC.

### **2.5.3. ΧΕΙΡΙΣΜΟΣ ΣΦΑΛΜΑΤΩΝ**

Για τον έλεγχο λαθών στη διάρκεια μιας SOAP μετάδοσης υπάρχει το <SOAP:fault>, που περιέχεται εντός του SOAP Body. Το SOAP:fault αποτελείται από τα υποχρεωτικά στοιχεία: <faultcode>: προσδιορίζει πού συνέβη το σφάλμα, <faultstring>: περιγράφει πώς δημιουργήθηκε το σφάλμα και τα προαιρετικά στοιχεία: <faultactor> και <detail> που περιέχουν περαιτέρω πληροφορίες για το σφάλμα.

Ανάλογα με την τιμή του <faultcode> διαμορφώνονται δύο κατηγορίες σφαλμάτων:

- <soap:Server>: λάθη την ευθύνη των οποίων δεν έχει ο πελάτης και δεν μπορούν να διορθωθούν από αυτόν.
- <soap:Client>: λάθη τα οποία οφείλονται στο περιεχόμενο του μηνύματος που έστειλε ο πελάτης-client .

Σε περίπτωση σφάλματος λοιπόν αποστέλλεται πίσω στην καλούσα διεργασία/σύστημα ένα SOAP Response μήνυμα στο οποίο

περιέχεται το πιο πάνω tag. Το tag αυτό περιλαμβάνει υποστοιχεία για την SOAP εξαίρεση που προέκυψε.

#### **2.5.4. ΠΑΡΑΔΕΙΓΜΑ**

Ακολουθεί ένα SOAP μήνυμα:

```
<soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <getProductDetails xmlns="http://warehouse.example.com/ws">
      <productID>827635</productID>
    </getProductDetails>
  </soap:Body>
</soap:Envelope>
```

Παραθέτουμε τον πιθανό τρόπο απάντησης μιας Διαδικτυακής Υπηρεσίας.

```
<soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <getProductDetailsResponse
xmlns="http://warehouse.example.com/ws">
      <getProductDetailsResult>
        <productName>Toptimate 3-Piece Set</productName>
        <productID>827635</productID>
        <description>3-Piece luggage set. Black
Polyester.</description>
        <price>96.50</price>
        <inStock>true</inStock>
      </getProductDetailsResult>
```

```
</getProductDetailsResponse>
</soap:Body>
</soap:Envelope>
```

#### **2.5.5. ΑΠΟΔΟΣΗ**

*Τα πρωτόκολλα CORBA και DCOM χρησιμοποιούν δυαδική κωδικοποίηση για τα μηνύματα που ανταλλάσσουν και θεωρούν ότι τα δύο συνδιαλεγόμενα συστήματα συμφωνούν ως προς τη μορφή των δεδομένων που ανταλλάσσουν. Εν αντιθέσει, το SOAP χρησιμοποιεί XML κωδικοποίηση στα μηνύματά του ούτως ώστε να αποδίδεται το κατάλληλο νόημα σε αυτά. Το γεγονός αυτό το καθιστά εύκολο στη διαχείριση, διαλειτουργικό και επεκτάσιμο, μολονότι υστερεί στην απόδοση συγκριτικά με τα δύο προηγούμενα. Ενδεικτικά αναφέρουμε ότι για το πρωτόκολλο SOAP έχουν μετρηθεί 500 μηνύματα/sec όταν πελάτης και διακομιστής βρίσκονται στο ίδιο μηχάνημα και 300 μηνύματα/sec στην αντίθετη περίπτωση.*

*Το μεγαλύτερο όμως πλεονέκτημα του SOAP είναι ότι μπορεί να διασύνδεει συστήματα που χρησιμοποιούν διαφορετικά middleware σε μία ενιαία πλατφόρμα επικοινωνίας, η οποία είναι ανεξάρτητη από το λογισμικό που τρέχει σε κάθε σύστημα. Η συμβατότητα είναι πολύ σημαντικό χαρακτηριστικό του SOAP και επιπλέον οι υπάρχουσες υποδομές middleware δεν είναι απαραίτητο να καταργηθούν. Κλείνοντας πρέπει να σημειώσουμε ότι το SOAP δεν αποτελεί στάσιμη τεχνολογία αλλά διαρκώς ανανεώνεται.*

## **2.6. WSDL (WEB SERVICES DESCRIPTION LANGUAGE)**

### **2.6.1. ΕΙΣΑΓΩΓΗ**

*Η* Web Services Description Language (WSDL) αναπτύχθηκε από την IBM σε συνεργασία με την Microsoft και πρόκειται για μια γλώσσα που βασίζεται στην XML. Η WSDL είναι μία πρότυπη γλώσσα περιγραφής των δομικών στοιχείων που συνθέτουν μία υπηρεσία διαδικτύου. Στόχος της είναι ο καθορισμός του τρόπου περιγραφής των Διαδικτυακών Υπηρεσιών.

### **2.6.2. ΠΕΡΙΓΡΑΦΗ**

*Μια* WSDL περιγραφή αποτελείται από δύο τμήματα:

#### *1. Abstract*

- Περιγραφή των τύπων δεδομένων που αποστέλλονται σε μία αίτηση προς την υπηρεσία.
- Δομή των μηνυμάτων που αποστέλλονται προς την υπηρεσία.
- Επιμέρους λειτουργίες που παρέχονται από την υπηρεσία.
- Οργάνωση σχετικών λειτουργιών σε επιμέρους διαπροσωπείες που προσφέρονται από την υπηρεσία.

#### *2. Concrete*

Περιγραφή μίας ή περισσότερων υπηρεσιών που προσφέρουν στιγμιότυπα των διαπροσωπειών που ορίζονται στο abstract κομμάτι.

- Συσχέτιση των διαπροσωπειών με κάποια συγκεκριμένα πρωτόκολλα επικοινωνίας, όπως το SOAP.

- Στιγμιότυπα της κάθε διαπροσωπείας, τα οποία προσφέρονται σε συγκεκριμένες διευθύνσεις (URLs).
- Στιγμιότυπα υπηρεσιών, ορισμένα σαν συλλογές από στιγμιότυπα διαπροσωπειών.

Πιο συγκεκριμένα, κάθε Διαδικτυακή Υπηρεσία συνδέεται με ένα αρχείο με επέκταση .wsdl στο οποίο καταγράφονται οι λειτουργίες που προσφέρει η συγκεκριμένη Διαδικτυακή Υπηρεσία, οι διαθέσιμες μέθοδοί της, τα πρωτόκολλα που μπορούν να χρησιμοποιηθούν για την κλήση της, η τοποθεσία της στο δίκτυο και τέλος ο τρόπος αναπαράστασης των δεδομένων εισόδου και εξόδου. Το αρχείο αυτό είναι ένα XML αρχείο το οποίο περιέχει ετικέτες με σημαντικές πληροφορίες.

Οι ετικέτες αυτές είναι:

- <definition>: Βρίσκεται στην κορυφή της ιεραρχίας και ορίζει μία ή περισσότερες Διαδικτυακές Υπηρεσίες.
- <type>: Πρόκειται για προαιρετική ετικέτα και περιέχει δηλώσεις σύνθετων τύπων δεδομένων σε περίπτωση που χρησιμοποιούνται στη Διαδικτυακή Υπηρεσία.
- <message>: Περιγράφει σε μορφή μηνυμάτων την πληροφορία που μεταφέρεται από τον πελάτη στον διακομιστή της Υπηρεσίας και το αντίστροφο.
- <portType>: Περιέχει μία ή περισσότερες ενέργειες, δηλαδή μηνύματα εισόδου/εξόδου που είναι δηλωμένα στο αρχείο .wsdl από την Διαδικτυακή Υπηρεσία.
- <operation>: Πρόκειται για μια ενέργεια, όπως περιγράψαμε στο <portType>.
- <binding>: Αναφέρεται σε ετικέτα <portType> η οποία έχει αναπτυχθεί με συγκεκριμένο πρωτόκολλο (π.χ. SOAP).

- `<service>`: Καθορίζει την τοποθεσία εύρεσης (URL) της Διαδικτυακής Υπηρεσίας.

Η WSDL περιγράφει μια Web Service σε δυο μέρη. Το «αφηρημένο» (abstract definition), που καθορίζει τη διεπαφή (interface) της υπηρεσίας και είναι ανεξάρτητο πλατφόρμας και γλώσσας προγραμματισμού και το «σταθερό» (concrete definition), το οποίο περιέχει λεπτομέρειες για την πρόσβαση στις υπηρεσίες.

### **2.6.3. ΠΑΡΑΔΕΙΓΜΑ**

Ακολουθεί ένα παράδειγμα σε WSDL. Υπάρχει κάποιος προμηθευτής που ονομάζεται etailer, προσδιορίζεται από το namespace "etailer". Η οντότητα αυτή περιλαμβάνει τον τύπο (portType) EShop με μια μόνο λειτουργία, την keywordSearch με είσοδο KSRequest και έξοδο το Product. Η KSRequest περιλαμβάνει δύο στοιχεία: μια λέξη κλειδί (keyword) και μία προαιρετική κατηγορία που μπορεί να πάρει μία από τις προκαθορισμένες τέσσερις τιμές. Το Product περιλαμβάνει τρία στοιχεία: το αναγνωριστικό (id), την κατηγορία και το salesrank.

```
<definitions targetNamespace="etailer">
  <portType name="EShop">
    <operation name="keywordSearch">
      <inputpart name="request" type="KSRequest"/>
      <outputpart name="product" type="Product"/>
    </operation>
  </portType>
  <complexType name="KSRequest">
    <element name="keyword" type="string"/>
    <element name="category" type="Category" minOccurs="0"/>
  </complexType>
  <complexType name="Product">
```



```

<element name="id" type="string"/>
<element name="category" type="Category"/>
<element name="salesrank" type="int" minOccurs="0"/>
</complexType>
<simpleType name="Category">
<restriction base="string">
<enumeration values="All, Books, Music, Movies"/>
</restriction>
</simpleType>
</definitions>

```

## **2.7. UDDI (UNIVERSAL DESCRIPTION , DEFINITION AND INTEGRATION)**

### **2.7.1. ΕΙΣΑΓΩΓΗ**

Το UDDI (Universal Description, Definition and Integration) είναι πρωτόκολλο καταχώρησης για Διαδικτυακές Υπηρεσίες. Προσφέρει την υποδομή για τη δημοσίευση και ανάκληση των Διαδικτυακών Υπηρεσιών με τη χρήση συγκεκριμένου API.

Σε γενικές γραμμές ικανοποιεί δύο στόχους:

Πρώτον, βοηθά στη δημιουργία προγραμμάτων πελατών (clients) που αλληλεπιδρούν με τις Διαδικτυακές Υπηρεσίες.

Δεύτερον, δίνει τη δυνατότητα του «binding», δηλαδή επιτρέπει στα προγράμματα-εφαρμογές να χρησιμοποιούν υπηρεσίες ώστε να αποκτούν πρόσβαση στις Web services που καλούν.

### **2.7.2. ΠΕΡΙΓΡΑΦΗ**

Υπάρχουν διακομιστές UDDI που επιτυγχάνουν τη δημοσίευση και την ανάκληση των Διαδικτυακών Υπηρεσιών. Οι διακομιστές αυτοί χωρίζονται σε κατηγορίες, οι οποίες είναι:

- Διακομιστές δημοσίας χρήσης: Επιτρέπουν στον καθένα να αναζητήσει ή να δημοσιεύσει δικές του Υπηρεσίες.
- Προστατευμένοι διακομιστές: Οι διακομιστές αυτοί αφορούν σε ερευνητικές ομάδες και επιτροπές του Διαδικτύου.
- Διακομιστές ιδιωτικής χρήσης: Οργανώνουν τις υπηρεσίες μιας εταιρίας ή ενός οργανισμού, προορίζονται δηλαδή για ενδοεπιχειρησιακούς σκοπούς.

Κάθε διακομιστής μπορεί να χειρίζεται τις παρακάτω κλάσεις αντικειμένων:

- businessEntity: Απευθύνεται στον κάτοχο της εκάστοτε Διαδικτυακής Υπηρεσίας.
- businessService: Αφορά στις υπηρεσίες που παρέχει μία επιχείρηση.
- bindingTemplate: Το αντικείμενο αυτό αντιστοιχίζεται σε κάθε Υπηρεσία και περιέχει γενικές πληροφορίες για τον τρόπο κλήσης της, το διακομιστή φιλοξενίας και κάποια περιγραφή αυτής.
- tModel: ``Δείχνει`` σε έναν δείκτη στο wsdl αρχείο της Υπηρεσίας.

Κάθε καταχώρηση στο UDDI περιέχει το αρχείο wsdl της υπηρεσίας και τη διεύθυνση στην οποία βρίσκεται η υπηρεσία στο Διαδίκτυο. Επιπροσθέτως, σε κάθε καταχώρηση υπάρχουν και άλλες

πληροφορίες που σχετίζονται με τον κάτοχο της υπηρεσίας και την πολιτική χρήσης της.

Επιπλέον, υποστηρίζεται υπηρεσία καταγραφής UDDI (UDDI registry) για κατηγοριοποίηση των πληροφοριών. Τέτοιες πληροφορίες μπορούν να είναι το όνομα του ιδιοκτήτη της Υπηρεσίας, η πολιτική λειτουργίας της κλπ. Η αναζήτηση στο UDDI registry γίνεται βάσει συγκεκριμένων χαρακτηριστικών της Υπηρεσίας.

Προκειμένου να δημοσιευθεί μια Διαδικτυακή Υπηρεσία αρχικά θα πρέπει να δημιουργηθεί ένας λογαριασμός σε κάποιον UDDI διακομιστή. Η ανάκληση της Υπηρεσίας μπορεί να γίνεται είτε στατικά κατά τη διάρκεια της κωδικοποίησης είτε δυναμικά κατά την διάρκεια εκτέλεσης μιας εφαρμογής. Πριν από τη χρήση μιας Υπηρεσίας πραγματοποιείται αναζήτηση και αξιολόγηση μέσω ενός UDDI διακομιστή.

## **2.8. ΑΣΦΑΛΕΙΑ ΤΩΝ XML ΤΕΧΝΟΛΟΓΙΩΝ ΣΤΙΣ WEB SERVICES**

Για την επίτευξη της ασφαλούς ανταλλαγής πληροφοριών ανάμεσα στον πελάτη και στο διακομιστή κατά τη διάρκεια της χρήσης μιας Διαδικτυακής Υπηρεσίας χρησιμοποιούνται κάποιοι από τους παρακάτω τρόπους:

- *Σύνδεση με http:* Υπάρχει όνομα και κωδικός χρήστη για την χρήση του διακομιστή. Τα δύο αυτά στοιχεία κρυπτογραφούνται και στη συνέχεια αποστέλλονται. Το πρόβλημα που εντοπίζεται εδώ αφορά στο ότι το κύριο μέρος του SOAP μηνύματος δεν κρυπτογραφείται.

- *Σύνδεση με HTTPs*: Εδώ γίνεται χρήση ψηφιακών πιστοποιητικών και του πρωτοκόλλου ασφαλείας SSL προκειμένου η μετάδοση να είναι περισσότερο ασφαλής. Για να επιτευχθεί αυτός ο τρόπος ασφάλειας γίνεται σύνδεση στη θύρα 443 κι όχι στην 80 που συνδέονται οι απλές HTTP συνδέσεις. Το κύριο πρόβλημα εδώ είναι ότι το HTTP δεν εγγυάται ότι το SOAP μήνυμα θα παραμείνει αναλλοίωτο κατά τη μεταφορά του στην περίπτωση που πρέπει να περάσει από κάποιον τρίτο ενδιάμεσο κόμβο που δεν χρησιμοποιεί HTTP.
- *XML Signature*: Πρόκειται για ένα XML αρχείο που αποτελεί ψηφιακή υπογραφή. Με τη χρήση αυτής της υπογραφής μπορούν να πιστοποιηθούν αρχεία οποιουδήποτε τύπου. Οι αλγόριθμοι που χρησιμοποιούνται είναι ο DSS και ο SHA-1.
- *XML Encryption*: Πρότυπο κρυπτογράφησης που δίνει τη δυνατότητα κρυπτογράφησης ανεξάρτητων τμημάτων ενός εγγράφου.
- *XKMS*: Πρότυπο διαχείρισης των δημόσιων και ιδιωτικών κλειδιών για τις επικοινωνίες των Διαδικτυακών Υπηρεσιών.
- *WS Security*: Πρότυπο ασφάλειας που αναπτύχθηκε από την IBM, την Microsoft και την Verisign και χρησιμοποιεί πολλές από τις γνωστές μεθόδους κρυπτογράφησης και αυθεντικοποίησης.

## ΚΕΦΑΛΑΙΟ 3 : Η ΤΕΧΝΟΛΟΓΙΑ XML

### 3.1. ΟΡΙΣΜΟΣ – ΠΕΡΙΓΡΑΦΗ

*Η XML (eXtensible Markup Language) αναπτύχθηκε από το W3C's XML Working Group το 1996. Συνδυάζει την ισχύ και την επεκτασιμότητα της SGML (Standard Generalized Markup Language), από την οποία προέρχεται, με την απλότητα που απαιτεί το Διαδίκτυο. Είναι ένα βιομηχανικό πρότυπο, ένας τρόπος αναπαράστασης δεδομένων ανεξάρτητος από συγκεκριμένα συστήματα καθώς και μια ευρέως υποστηριζόμενη και ανοικτή τεχνολογία για την περιγραφή δεδομένων.*

*Δεν πρόκειται για μια γλώσσα προγραμματισμού αλλά για μία δηλωτική γλώσσα που ακολούθησε την ASN1. Κύρια διαφορά τους είναι ότι στην XML χρησιμοποιούνται αποκλειστικά χαρακτήρες (character based), ενώ η ASN1 χρησιμοποιεί bit. Επιπλέον η δομή της είναι απλή και εύκαμπτη. Όπως και η HTML (HyperText Markup Language, γλώσσα σήμανσης υπερκειμένου), η XML εσωκλείει τα δεδομένα της σε ετικέτες, παρόλα αυτά υπάρχουν διαφορές ανάμεσα στις δύο τεχνολογίες. Η πρώτη διαφορά έγκειται στο ότι οι ετικέτες στην XML αφορούν στην έννοια του εσωκλειόμενου κειμένου, ενώ οι HTML ετικέτες προσδιορίζουν τον τρόπο παρουσίασης του εσωκλειόμενου περιεχομένου της ετικέτας. Η δεύτερη διαφορά αφορά στον τρόπο χρήσης των ετικετών. Με την XML μπορούμε να γράψουμε τις ετικέτες μας για να περιγράψουμε το περιεχόμενο τους σε έναν ιδιαίτερο τύπο εγγράφου. Σε αντίθεση, με την HTML είμαστε περιορισμένοι γιατί μπορούμε να χρησιμοποιήσουμε μόνο τις προκαθορισμένες HTML ετικέτες.*

Από μια άλλη οπτική γωνία, η XML είναι ένα τυποποιημένο κείμενο για την αναπαράσταση δομημένης πληροφορίας στο Διαδίκτυο. Δεν είναι απαραίτητο, επομένως, να είναι κάποιος προγραμματιστής για να μπορέσει να την χρησιμοποιήσει ή να τη μάθει. Διευκολύνει τον υπολογιστή να παράγει δεδομένα, να διαβάζει δεδομένα και να εξασφαλίζει τη σαφήνεια της δομής των δεδομένων.

Σε γενικές γραμμές πρόκειται για απλή δηλωτική γλώσσα με εύκαμπτη δομή που χρησιμοποιείται για τη δόμηση δεδομένων. Με την έννοια δομημένα δεδομένα εννοούμε μία συλλογή δεδομένων όπως είναι για παράδειγμα τα λογιστικά φύλλα, οι κατάλογοι διευθύνσεων, οι οικονομικές συναλλαγές κλπ. Η XML είναι, δηλαδή, ένα σύνολο κανόνων (ή διαφορετικά ένα πακέτο κατευθυντήριων γραμμών ή συμβάσεων) για το σχεδιασμό μορφών κειμένου οι οποίες διευκολύνουν τη δόμηση των δεδομένων.

### **3.2. ΠΛΕΟΝΕΚΤΗΜΑΤΑ ΧΡΗΣΗΣ**

Μερικοί από τους λόγους για τους οποίους η XML έχει γίνει ευρέως αποδεκτή είναι:

- Σκέτο κείμενο: Δεδομένου ότι η XML δε χρησιμοποιεί τη δυαδική μορφή, αλλά αλφαριθμητική μορφή, μπορούμε να δημιουργήσουμε και να επιμεληθούμε τα XML αρχεία με οποιοδήποτε πρόγραμμα, όπως έναν πρότυπο συντάκτη κειμένου μέχρι και ένα οπτικό περιβάλλον ανάπτυξης.
- Προσδιορισμός δεδομένων: Η XML μας λέει το είδος δεδομένων έχουμε και όχι πώς να το παρουσιάσουμε.
- Δυνατότητα Υποστήριξης Στυλ – Ύφους (stylability): Υπάρχει η δυνατότητα υπαγόρευσης του τρόπου απεικόνισης των δεδομένων μέσω του προτύπου stylesheet XSL.

- Επαναχρησιμοποίηση κώδικα σε ένα – ίδιο – έγγραφο (inline reusability): Τα έγγραφα XML μπορούν να συντεθούν από χωριστές οντότητες, οι οποίες μπορούν να συμπεριληφθούν σε ένα in-line έγγραφο, κάτι που δεν συμβαίνει με την HTML.
- Συνδεσιμότητα: Υπάρχει η δυνατότητα δημιουργίας συνδέσεων μεταξύ εγγράφων.
- Ευκολία Επεξεργασίας: Η κανονική και συνεπής σημειογραφία της XML καθιστά ευκολότερη τη δημιουργία ενός προγράμματος επεξεργασίας των δεδομένων.
- Ιεραρχική δομή: Τα έγγραφα της XML παρουσιάζουν ιεραρχική δομή και έτσι προσπελαύνονται και κατηγοριοποιούνται πιο γρήγορα.
- Διατίθενται έτοιμοι τρόποι σύνδεσης των κειμένων XML με τα πλέον σύγχρονα προγραμματιστικά περιβάλλοντα, όπως το Document Object Model (DOM) και το Simple API for XML (SAX).
- Είναι επεκτάσιμη και ανεξάρτητη από πλατφόρμες, γεγονός που την καθιστά απρόσβλητη από τεχνολογικές αλλαγές.
- Είναι πλήρως συμβατή με Unicode, οπότε μπορεί να χειριστεί την πληροφορία που έχει γραφεί σε οποιαδήποτε ανθρώπινη γλώσσα. Παράλληλα, υποστηρίζει διεθνείς και τοπικές προσαρμογές.

### **3.3. ΧΑΡΑΚΤΗΡΙΣΤΙΚΑ**

#### **3.3.1. ΕΤΙΚΕΤΕΣ**

Η βασική δομή σε ένα XML έγγραφο είναι οι ετικέτες (tags), δηλαδή προσδιοριστικά που εσωκλείονται σε παρενθέσεις γωνία, όπως < .....>. Τα tags δεν ορίζονται από τον χρήστη. Ετικέτες υπάρχουν και στην HTML, (π.χ. το ζεύγος <B> </B> ή το μοναδικό

<Br>), όμως είναι προκαθορισμένες και σχετίζονται μόνο με την εμφάνιση του κειμένου. Παραθέτουμε ένα παράδειγμα δομής των ετικετών:

```
<message>
  <to>you@yourAddress.com</to>
  <from>me@myAddress.com</from>
  <subject>XML </subject>
  <text>
    Why do we use XML???...
  </text>
</message>
```

Οι ετικέτες κατά σειρά προσδιορίζουν: όλο το μήνυμα, τις διευθύνσεις προορισμού και αποστολής, το θέμα και το κείμενο του μηνύματος. Όπως και στην HTML η ετικέτα <to> (ετικέτα αρχής), πρέπει να ακολουθείται και από ετικέτα τέλους </to>. Τα δεδομένα ανάμεσα στις δύο αυτές ετικέτες προσδιορίζουν συγκεκριμένο XML δεδομένο.

*Παραθέτουμε ενδεικτικά ένα απλό XML αρχείο:*

```
<?xml version="1.0"?>
<catalog>
  <product partNum = " ABC – 0001" SaleDate ="12/2/2005">
    <productName>XXXXXX 2.02</productName>
    <price>1000 </price>
    <color> black </color>
  </product>
  <product partNum = " DEF – 0011" SaleDate ="12/1/2005">
    <productName>KKKKKKK 2.01</productName>
    <price>1500 </price>
    <color> red </color>
```



</product>

</catalog>

Στο έγγραφο αυτό μπορούμε να διακρίνουμε τις ετικέτες αλλά και ένα στοιχείο (element), το στοιχείο ρίζας, που περιλαμβάνει τα υπόλοιπα στοιχεία και είναι το πρώτο μετά τη δήλωση XML. Τα στοιχεία εντίθενται το ένα μέσα στο άλλο για να σχηματίσουν ιεραρχίες – με το στοιχείο ρίζας στην κορυφή της ιεραρχίας. Με την ιδιότητα αυτή της ενθυλάκωσης (nesting) παρέχεται η δυνατότητα να κατασκευαστούν στοιχεία με πιο πολύπλοκη εσωτερική δομή.

Κάθε στοιχείο (element) μπορεί να περιέχει κάποια χαρακτηριστικά (attributes), που είναι ζεύγη ονόματος – τιμής στις ετικέτες αρχής.

### **3.3.2 DTD**

Ένα αρχείο DTD – Document Type Definition - είναι ένα μοντέλο με το οποίο μπορούμε να ορίσουμε ρητά τη δομή μιας κατηγορίας XML εγγράφων. Κληρονομήθηκε από τη γλώσσα SGML και χρησιμοποιείται για να περιγράψει κάθε αντικείμενο σε ένα XML έγγραφο, όπως τα στοιχεία και τα χαρακτηριστικά. Αποτελεί δηλαδή ένα σύνολο κανόνων που αφορούν στα tags, ακόμα ορίζονται κανόνες για το ποια ονόματα στοιχείων και υπο-στοιχείων είναι επιτρεπτά καθώς και για τον τύπο των χαρακτηριστικών του κάθε στοιχείου. Για καλύτερη κατανόηση ακολουθεί ένα DTD, για έναν κατάλογο ζώων, που ορίζει ότι το κάθε ζώο έχει ένα όνομα, ανήκει σε ένα συγκεκριμένο είδος και παράγει συγκεκριμένο ήχο:

<!ELEMENT animal – list (animal)\*>

<!ELEMENT animal (name,type,sound)>

<!ELEMENT name (#PCDATA)>

```
<!ELEMENT type (#PCDATA)>
<!ELEMENT sound (#PCDATA)>
```

*Ένα XML έγγραφο που ακολουθεί τους κανόνες του DTD είναι:*

```
<?xml version="1.0"?>
<!DOCTYPE animal – list SYSTEM “zoology.dtd”>
<animal-list>
<animal>
<name>Lola</name>
<type>cat</type>
<sound>niaou</sound>
</animal>
<animal>
<name>Jack</name>
<type>dog</type>
<sound>gav</sound>
</animal.>
</animal-list>
```

### **3.3.3 ΣΧΗΜΑ XML ( XML SCHEMA )**

*Αν και τα DTD είναι τα καθιερωμένα πρότυπα για τον ορισμό εγγράφων XML έχουν σοβαρούς περιορισμούς. Τα DTD, όπως αναφέραμε έχουν τις ρίζες τους στην SGML, κάτι που σημαίνει ότι αρχικά σχεδιάστηκαν για τον ορισμό γλωσσών χαρακτηρισμού εγγράφων και όχι για τη δημιουργία σχημάτων βάσεων δεδομένων. Επιπλέον δεν παρέχουν επαρκή έλεγχο στο περιεχόμενο των στοιχείων. Σαν αποτέλεσμα των περιορισμών των DTD και της αυξανόμενης χρήσης της XML αναπτύχθηκε η γλώσσα ορισμού σχημάτων XML Schema Definition (XSD). Η προδιαγραφή για την XSD υπάρχει στην διεύθυνση: [www.w3.org/XML/Schema.html](http://www.w3.org/XML/Schema.html)*

σύσταση του οργανισμού W3C. Τα XML Schemas σχεδιάστηκαν για τον ίδιο σκοπό με αυτόν των DTD: τον ορισμό κατηγοριών XML εγγράφων. Η βασική διαφορά τους είναι ότι τα σχήματα XML διαχωρίζουν τα στοιχεία σε τύπους, όπως πολύπλοκα στοιχεία και απλά αντίστοιχα. Ακόμα μπορούν να υποστούν χειρισμούς, παραδείγματος χάρη μπορούν να προστεθούν ή να διαγραφούν στοιχεία, όπως σε οποιοδήποτε άλλο έγγραφο XML. Οι ιδιότητες ανήκουν πάντα σε απλούς τύπους και ουσιαστικά είναι έγγραφα XML που καθορίζουν πώς πρέπει να δομούνται κάποια άλλα έγγραφα XML. Το πλεονέκτημά τους σε σχέση με τα DTDs (Document Type Definitions) είναι ότι μπορούν να υποστούν χειρισμούς, παραδείγματος χάρη μπορούν να προστεθούν ή να διαγραφούν στοιχεία, όπως σε οποιοδήποτε άλλο έγγραφο XML.

Τα XML Schemas εγγυώνται ότι σε κάθε στοιχείο αποθηκεύεται ο σωστός τύπος δεδομένων. Επίσης, επιτρέπουν στον χρήστη να ορίσει δικούς του τύπους δεδομένων, διαφορετικούς από τους ήδη υπάρχοντες στο σύστημα τύπων XML Schema Definition (XSD). Διευκολύνουν τη μετάβαση ανάμεσα σε πλατφόρμες διότι το σύστημα τύπων XML Schema Definition (XSD) είναι ανεξάρτητο από πλατφόρμες.

Παραθέτουμε ένα έγγραφο XML με το αντίστοιχο XML Schema:

(catalog.xsd)

```
<xsd:schema xmlns:xsd = "http://www.w3.org/1999/XMLSchema">
<xsd:element name = "Catalog" type = "catalogType"/>
<xsd:complexType name = "CatalogType">
<xsd:element name = "product" type = "ProductType"/>
<xsd:attribute name = "onSaledate" type = "xsd:date"/>
<xsd:attribute name = "partNum" type = "Sku"/>
</xsd:complexType>
```

```

<xsd:complexType name = "ProductType">
  <xsd:element name = "productName" type = "xsd:string">
  <xsd:element name = "price" type = "xsd:decimal"/>
  <xsd:element name = "description" type = "xsd:string" minOccurs = "0"/>

  <xsd:simpleType name = "Sku" base = "xsd:string">
  <xsd:pattern value = "\ [a-z] (3) - [a-z] (3) d(3) "/>
  </xsd:simpleType>
</xsd:schema>

```

(catalog.xml)

```

<?xml version="1.0"?>
<catalog>
  <product partNum = " ABC - Pro336" onSaleDate ="12/2/2005">
    <productName>XXXXXX 2.02</productName>
    <price>1000 </price>
    <description> ..... </description>
    <color> black </color>
  </product>
  <product partNum = " DEF - PRO343" onSaleDate ="12/1/2005">
    <productName>KKKKKKK 2.01</productName>
    <price>1500 </price>
    <description> ..... </description>
    <color> red </color>
  </product>
</catalog>

```

### **3.3.4. ΟΡΘΟΤΗΤΑ XML ΕΓΓΡΑΦΟΥ**

Για να είναι ένα XML έγγραφο σωστό θα πρέπει να είναι συγχρόνως και :

- Καλοσχηματισμένο (well-formed), δηλαδή να υπακούει σε όλους τους συντακτικούς κανόνες της XML. Εάν ένα έγγραφο δεν είναι συντακτικά σωστό, τότε δεν θεωρείται έγγραφο XML, η ανάλυση (parsing) σταματά και ο αναλυτής (parser) δίνει σφάλμα.
- Έγκυρο (valid), δηλαδή να ικανοποιεί ένα σύνολο κανόνων που έχει ορίσει ο χρήστης, οι οποίοι θα περιλαμβάνεται σε προαιρετικά έγγραφα XML, όπως οι Ορισμοί Τύπων Εγγράφων (Document Type Definitions–DTDs) και τα Σχήματα XML (XML Schemas).

### **3.4. ΤΕΧΝΟΛΟΓΙΕΣ XML**

Η XML συνδυάζει διαφορετικές τεχνολογίες. Πιο συγκεκριμένα, η XML 1.0 ορίζει τις "ετικέτες" και τα «γνώρίσματα». Πέρα από την XML 1.0, «η οικογένεια XML» είναι ένα διαρκώς αναπτυσσόμενο σύνολο λειτουργικών μονάδων οι οποίες προσφέρουν χρήσιμες υπηρεσίες για την ολοκλήρωση σημαντικών έργων που προκύπτουν. Τα XPointer και τα XFragments υποδεικνύουν θέσεις ενός εγγράφου XML. Το XPointer λειτουργεί σαν URL αλλά αντί να υποδεικνύει έγγραφα στον Ιστό, υποδεικνύει κομμάτια πληροφοριών ενός εγγράφου XML. Η Xlink περιγράφει έναν προκαθορισμένο τρόπο εισαγωγής υπερσυνδέσμων (hyperlinks) σε αρχεία XML. Το XSL είναι προηγμένη γλώσσα μορφοποίησης σελίδων. Βασίζεται στο XSLT, πρόκειται για γλώσσα μετασχηματισμού η οποία χρησιμοποιείται για αναδιάταξη, πρόσθεση και διαγραφή ετικετών

και γνωρισμάτων. Το DOM είναι ένα προκαθορισμένο σύνολο λειτουργιών για τη διαχείριση αρχείων XML (και HTML) σε συγκεκριμένη γλώσσα προγραμματισμού. Τα XML Schemas 1.2 επιτρέπουν στους κατασκευαστές λογισμικού να ορίζουν με ακρίβεια τις δομές των δικών τους XML εγγράφων. Τέλος πρέπει να σημειώσουμε ότι υπάρχουν και άλλα εργαλεία τα οποία βρίσκονται υπό διαμόρφωση ή είναι ήδη διαθέσιμα.

### **3.5. XML ΚΑΙ HL7**

#### **3.5.1. ΙΣΤΟΡΙΚΗ ΑΝΑΔΡΟΜΗ**

Από το 1993, η Ευρωπαϊκή Επιτροπή Προτυποποίησης (CEN) ξεκίνησε να μελετά πολλούς τρόπους σύνταξης μηνυμάτων (συμπεριλαμβανομένων των ANS.1, ASTM, EDIFACT, EUCLIDES και ODA) για τη χρήση τους ως πρότυπο για την ανταλλαγή μηνυμάτων σε οργανισμούς υγείας. Μια αναφορά που ακολούθησε επέκτεινε την μελέτη της CEN για να συμπεριλάβει και την SGML (πρόγονο της XML). Χρησιμοποιώντας την ίδια μεθοδολογία, τα ίδια παραδείγματα, τα ίδια μοντέλα δεδομένων και τους ίδιους συντελεστές αξιολόγησης, φάνηκε η υπεροχή της SGML συγκριτικά με όλες τις άλλες γλώσσες.

Το Φεβρουάριο του 1998, αναπτύχθηκε η XML από τον World Wide Web Consortium (W3C) και δοκιμάστηκε επιτυχώς σαν γλώσσα σύνταξης μηνυμάτων HL7 των εκδόσεων 2.x και 3.

Το 1999 το XML SIG ανέπτυξε ένα έγγραφο σε συνεργασία με το Control/Query TC με τίτλο: “HL7 Recommendation: Using XML as a Supplementary Messaging Syntax for HL7 Version 2.3.1 – HL7 XML Special Interest Group, Informative Document” και εγκρίθηκε σαν

πληροφοριακό HL7 έγγραφο σε επίπεδο μελών τον Φεβρουάριο του 2000.

Τέλος, τον Αύγουστο του 2000 πραγματοποιήθηκε μία συνάντηση του HL7 συμβουλίου στη Δρέσδη όπου αποφασίστηκε η XML να γίνει η δεύτερη επίσημη γλώσσα κωδικοποίησης για τις εκδόσεις 2.3.1 και 2.4 του HL7 αλλά και για όλες τις μελλοντικές του 2.x εκδόσεις.

### **3.5.2. ΚΑΝΟΝΕΣ ΓΙΑ ΤΑ XML ΜΗΝΥΜΑΤΑ**

Θεωρητικά πολλές XML κωδικοποιήσεις θα μπορούσαν να χρησιμοποιηθούν σαν εναλλακτικοί τρόποι σύνταξης μηνυμάτων της έκδοσης 2.x του HL7. Για το λόγο αυτό ο οργανισμός του HL7 εξέδωσε ένα έγγραφο που περιγράφει τον επίσημο τρόπο κωδικοποίησης που υιοθετήθηκε ως πρότυπο από τον οργανισμό. Περιγράφει κυρίως την μετάφραση των μηνυμάτων από την HL7 2.x κωδικοποίηση σε XML μορφή. Η μετατροπή γίνεται βάσει συγκεκριμένων αρχών και κανόνων.

Οι κανόνες αυτοί αναπαριστούν τις δομές μηνυμάτων (message structures) HL7 σαν XML elements. Κάθε δομή μηνύματος περιέχει τμήματα (segments) που επίσης αναπαριστώνται σαν XML elements. Τα τμήματα περιέχουν πεδία (fields) και αυτά αναπαριστώνται σαν elements. Ο τύπος δεδομένων (data type) ενός πεδίου αποθηκεύεται ως καθορισμένη ιδιότητα στη λίστα ιδιοτήτων (attributes list) του πεδίου. Άλλες σταθερές ιδιότητες χρησιμοποιούνται για να επεκταθούν συντομογραφίες και για να καθοριστούν περιορισμοί σε τιμές πινάκων του HL7.

Ολόκληρο το HL7 πρότυπο είναι καταγεγραμμένο σε μια βάση δεδομένων η οποία περιέχει τον επίσημο ορισμό των γεγονότων (events), των μηνυμάτων (messages), των τμημάτων (segments), των πεδίων (fields), των τύπων δεδομένων (data types), των πινάκων (tables) και των αντίστοιχων τιμών τους (table values). Η βάση ανανεώνεται διαρκώς ώστε να αντανakλά επακριβώς το περιεχόμενο των επίσημων εγγράφων του προτύπου και να περιλαμβάνει τις κατά καιρούς εγκεκριμένες από τα μέλη προτάσεις και διορθώσεις. Μέσα στην βάση δεδομένων του HL7 όλα τα δεδομένα ελέγχονται για τη συνέπειά τους.

### **3.5.3. ΠΑΡΑΔΕΙΓΜΑ**

Πιο κάτω παραθέτουμε ένα παράδειγμα ενός HL7 μηνύματος στην παραδοσιακή του μορφή:

```
MSH|^~\&|LAB^foo^bar|767543|ADT|767543|19900314130405||ACK  
^|XX3657|P|2.3.1<CR> MSA|AA|ZZ9380<CR>
```

Ακολουθεί το ίδιο μήνυμα γραμμένο σε XML:

```
<!DOCTYPE ACK SYSTEM "hl7_v231.dtd">  
<ACK>  
<MSH>  
<MSH.1>|</MSH.1>  
<MSH.2>^~\&|</MSH.2>  
<MSH.3>  
<HD.1>LAB</HD.1>  
<HD.2>foo</HD.2>  
<HD.3>bar</HD.3>  
</MSH.3>  
<MSH.4><HD.1>767543</HD.1></MSH.4>  
<MSH.5><HD.1>ADT</HD.1></MSH.5>
```



```

<MSH.6><HD.1>767543</HD.1></MSH.6>
<MSH.7>19900314130405</MSH.7>
<MSH.9><CM_MSG_TYPE.1>ACK</CM_MSG_TYPE.1></MSH.9>
<MSH.10>XX3657</MSH.10>
<MSH.11><PT.1>P</PT.1></MSH.11>
<MSH.12><VID.1>2.3.1</VID.1></MSH.12>
</MSH>
<MSA>
<MSA.1>AA</MSA.1>
<MSA.2>ZZ9380</MSA.2>
</MSA>
</ACK>

```

Όπως ισχύει πάντοτε με τα XML έγγραφα όταν αυτά υφίστανται επεξεργασία από ένα validating parser, τα κενά διαστήματα μεταξύ των elements αγνοούνται και εξυπηρετούν μόνο την ευκολότερη ανάγνωση του XML αρχείου. Έτσι σε πραγματικά HL7 μηνύματα μπορούν να αγνοηθούν εντελώς εάν το μήκος του μηνύματος είναι μεγάλο.

#### **3.5.4. XML ΜΕΤΑΣΧΗΜΑΤΙΣΜΟΙ**

Στις προηγούμενες παραγράφους παρουσιάσαμε συνοπτικά τη γλώσσα XML. Η XML όμως σαν γλώσσα δεν παρέχει στο χρήστη τη δυνατότητα να μορφοποιήσει τον τρόπο παρουσίασης των δεδομένων προκειμένου να καθίστανται πιο εύχρηστα. Για το στυλ και την μορφοποίηση των εγγράφων XML, υπάρχουν τεχνολογίες που καθιστούν τα έγγραφα XML σε μεγάλο βαθμό παραμετροποιήσιμα σε ότι έχει να κάνει με την εμφάνιση τους.

Ένας τρόπος ελέγχου της μορφοποίησης των XML εγγράφων είναι να ενσωματώσουμε στο έγγραφο κάποια HTML tags, με στόχο την επιθυμητή παρουσίαση του εγγράφου μέσα από ένα web browser. Ειδικότερα αν καταφέρουμε να έχουμε μια αυστηρή έκδοση της HTML που να υπακούει σε όλους τους κανόνες δομής της XML, τότε θα μπορέσουμε να δημιουργήσουμε τη μορφοποίηση που θέλουμε, διατηρώντας όμως την αυστηρή δομή της XML. Μια τέτοια προσπάθεια εκφράζεται από την εξελιγμένη γλώσσα σήμανσης την XHTML.

Αναπτύχθηκαν στη συνέχεια και άλλες τεχνολογίες παράλληλες με την XML που προσφέρουν σημαντικές επιλογές στο χρήστη. Για την επεξεργασία των εγγράφων XML χρησιμοποιούνται και οι τεχνολογίες XSLT και XSLFO (XSL Formatting Objects), οι οποίες μαζί με την σύσταση xpath (XML path language) υπάγονται στην XSL (XML Stylesheet Language) του W3C. Η σύσταση XSL αφορά στο στυλ των εγγράφων XML και μας δίνει τη δυνατότητα να μετατρέπουμε έγγραφα XML σε άλλα έγγραφα XML με διαφορετικό DTD ή XML Schema ή σε έγγραφα διαφορετικών μορφών (HTML text και άλλα). Ακόμα η XSL, διευκολύνει τη δημοσίευση μεγάλου όγκου από έγγραφα και την αναδιοργάνωση εγγράφων XML (π.χ. δημιουργία πινάκων περιεχομένων ή άλλων χρηστικών δομών αναπαράστασης των πληροφοριών). Η σύσταση XSLT αφορά στο μετασχηματισμό ενός εγγράφου XML, ενώ η σύσταση XSLFO στη μορφοποίησή του για παρουσίαση. Την ίδια λειτουργία με την XSLFO επιτελεί και η τεχνολογία CSS (Cascading Style Sheets). Ωστόσο η CSS είναι δημοφιλέστερη και μονοπωλεί το ενδιαφέρον των κατασκευαστών λογισμικού και παρόχων περιεχομένου. Ακολουθεί αμέσως μια ανάλυση των τριών βασικών τεχνολογιών με τις οποίες επιτυγχάνεται ο μετασχηματισμός και η μορφοποίηση των εγγράφων XML με την XSL να αποτελείται από τρία μέρη:

- XSLT – Γλώσσα μετασχηματισμού XML αρχείων
- XPath – Γλώσσα για την πλοήγηση σε XML αρχεία
- XSL-FO- Γλώσσα για την μορφοποίηση XML αρχείων

## ο CSS

*Η* τεχνολογία CSS είναι ένας μηχανισμός κανόνων που χρησιμοποιείται για να προσδώσει συγκεκριμένο στυλ στα στοιχεία ενός εγγράφου. Με τη χρήση ενός αρχείου κανόνων CSS παρέχονται εντολές στον browser για την εμφάνιση ενός αρχείου XML ή HTML. Έτσι, διαχωρίζεται η εμφάνιση μιας σελίδας από το περιεχόμενο της. Το CSS συστάθηκε από το W3C και έχει δύο εκδόσεις: την έκδοση CSS 1.0 και την CSS 2.0 που υποστηρίζει την XML σε μεγαλύτερο βαθμό. Με βάση το CSS δημιουργήθηκε και η σύσταση XSLFO από το W3C.

## ο XSLFO

*Το* XSLFO είναι αρχείο XSL που περιέχει διαφορετικά tags μέσα σε κάθε στοιχείο <xsl:template>. Οι ετικέτες αυτές δεν είναι ίδιες με αυτές που υπάρχουν σε ένα XSLT stylesheet. Τα νέα tags που χρησιμοποιούνται στα XSLFO βασίζονται στη λογική των CSS stylesheets. Κανένας από τους γνωστούς browser δεν υποστηρίζει το XSLFO, οι περισσότεροι υποστηρίζουν πλέον εναλλακτικά το XSLT και το CSS2. Η συντριπτική πλειοψηφία των εγγράφων βασίζεται στην HTML και την XHTML για την εμφάνιση τους στο Web. Με αυτή τη λογική το XSLFO μπορεί να αποτελέσει μια τελείως νέα λογική, αλλά αυτό που χρησιμοποιείται στην πράξη και θα συνεχίσει να υποστηρίζεται στο μέλλον είναι ο συνδυασμός XSLT και XHTML, ως εξέλιξη της απλής HTML.

## ο XSLT

Όπως προαναφέραμε, η τεχνολογία του XSL μετασχηματισμού είναι η πλέον αποδεκτή γλώσσα μορφοποίησης και παρουσίασης αρχείων XML. Πρόκειται για τα αρχικά των λέξεων *eXtensible Stylesheet Language* και με λίγα λόγια μετατρέπει τα XML αρχεία σε HTML, ώστε να είναι κατάλληλα για παρουσίαση στο χρήστη με τη χρήση ενός browser. Αρχικά η XSLT είχε σχεδιαστεί για να υποστηρίζει την XSL-FO, όμως στην πορεία καθιερώθηκε ως γλώσσα μετατροπής κειμένων XML σε οποιαδήποτε μορφή κειμένου, αλλά και γενικότερα ως η προτιμώμενη τεχνολογία για ποικίλες μετατροπές. Το αρχείο xsl που θα μορφοποιήσει το xml αρχείο καλείται με την πρόσθεση κώδικα που δείχνει τη θέση του αρχείου xslt. Πιο συγκεκριμένα η κλήση του αρχείου γίνεται ως εξής:

```
<?xml-stylesheet type="text/xsl" href="MSH.xslt"?>
```

Παρατηρούμε ότι μετά το href ( εντολή αναφοράς ) έχουμε το όνομα του αρχείου xsl.

Ο λόγος δημιουργίας του XSL μετασχηματισμού ήταν ότι η XML δεν χρησιμοποιεί προκαθορισμένα αναγνωριστικά (tags), σε αντίθεση με την XSL που κάνει τη μετατροπή των XML αρχείων δίνοντας κάποια «σημασία» στα αναγνωριστικά τους. Ο μετασχηματισμός αυτός γίνεται με την χρήση της XSLT. Χρησιμοποιώντας το XPath για την πλοήγηση μέσα στο αρχείο, η XSLT μετατρέπει το XML αρχείο σε κάποιο άλλο (XML ή HTML ή XHTML ή text). Έτσι είμαστε σε θέση να προσθαφαιρέσουμε στοιχεία και γνωρίσματα στο αρχείο που θα παρουσιάσουμε και να καθορίσουμε τον τρόπο εμφάνισης του.

Τρία είναι τα βασικά στοιχεία του μετασχηματισμού που χρησιμοποιούνται στη δημιουργία των HL7 μηνυμάτων. Αναλυτικότερα έχουμε:

- **<xsl:template>:** Χρησιμοποιείται για τη δημιουργία προτύπων. Το αρχείο εξόδου, που μετατρέπεται έπειτα σε HTML, ορίζεται από το περιεχόμενο του template.
- **<xsl:value-of>:** Χρησιμοποιείται για την παραγωγή της τιμής του element του XML αρχείου, προκειμένου να χρησιμοποιηθεί στο αρχείο εξόδου. Μπορεί να το βρούμε και στη μορφή `<xsl:value-of select="catalog/cd/title"/>`.
- **<xsl:for-each>:** Πρόκειται για μια αρκετά χρήσιμη και σημαντική εντολή της XSL γιατί επιτυγχάνεται μέσω αναδρομής η δημιουργία δυναμικού πίνακα.
- **<xsl:apply-templates>:** Σκοπός της είναι η μορφοποίηση του element και όλων των υποστοιχείων του για τα οποία καλείται η εντολή αυτή.

#### ○ **Xpath**

Το Xpath χρησιμοποιεί εκφράσεις για την επιλογή κόμβων (nodes) ή συνόλου κόμβων (node-sets) μέσα σε ένα XML αρχείο. Στη συνέχεια θα παρουσιάσουμε συνοπτικά την ορολογία που χρησιμοποιεί το Xpath.

- **Nodes:** Στο XPath υπάρχουν 7 είδη κόμβων: element (στοιχείο), attribute (αναγνωριστικό), text (κείμενο), namespace (χώρος ονόματος), processing-instruction (εντολή επεξεργασίας), comment (σχόλιο) και document nodes (κόμβοι αρχείου).
- **Atomic Values:** Κόμβοι χωρίς παιδιά (children) ή γονείς (parents).

- **Items:** Είτε ατομικές τιμές είτε κόμβοι.

Οι κόμβοι στο XPath σχετίζονται με τους πιο κάτω τρόπους:

- **Parent (Γονέας):** Κάθε στοιχείο ή αναγνωριστικό έχει ένα γονέα.
- **Children (Παιδιά):** Κάθε στοιχείο μπορεί να έχει μηδέν, ένα ή περισσότερα παιδιά.
- **Siblings (Συγγενείς):** Κόμβοι που έχουν τον ίδιο γονέα.
- **Ancestors:** Ο γονέας κάθε κόμβου, αναδρομικά.
- **Descendants:** Τα παιδιά κάθε κόμβου, αναδρομικά.

Η επιλογή των γνωστών κόμβων στο XPath γίνεται με τη χρήση εκφράσεων. Παρακάτω παρουσιάζουμε κάποιες χρήσιμες τέτοιες εκφράσεις.

Έκφραση	Περιγραφή
nodename	Επιλογή των παιδιών κάθε κόμβου
/	Επιλογή από τον αρχικό κόμβο
//	Επιλογή κόμβων από τον τρέχοντα κόμβο, οι οποίοι ταιριάζουν με την επιλογή ανεξαρτήτως της θέσης τους
.	Επιλογή του τρέχοντος κόμβου
..	Επιλογή του γονέα του τρέχοντος κόμβου
@	Επιλογή των αναγνωριστικών (attributes)

Η επιλογή των άγνωστων κόμβων γίνεται με τις παρακάτω εκφράσεις:

Ειδικός	Περιγραφή
---------	-----------

Χαρακτήρας	
*	Ταιριάζει κάθε κόμβο του στοιχείου
@*	Ταιριάζει κάθε κόμβο αναγνωριστικού
node()	Ταιριάζει κάθε κόμβο οποιουδήποτε τύπου

Βέβαια υπάρχουν και άλλες εκφράσεις που χρησιμοποιεί το XPath για να περιγράψει τη διαδρομή στο XML έγγραφο, αλλά αυτές δεν χρησιμοποιούνται στο μετασχηματισμό για την παρουσίαση των XML μηνυμάτων του HL7 και γι' αυτό δεν γίνεται σχετική αναφορά.

### **3.5.5. ΠΛΕΟΝΕΚΤΗΜΑΤΑ ΤΗΣ ΧΡΗΣΗΣ ΤΗΣ XML ΓΙΑ ΤΗΝ ΚΩΔΙΚΟΠΟΙΗΣΗ ΜΗΝΥΜΑΤΩΝ HL7**

Η XML επιλέχθηκε για την κωδικοποίηση των ανταλλασσόμενων HL7 μηνυμάτων ανάμεσα σε συνδιαλασσόμενες εφαρμογές για τους παρακάτω λόγους:

- *Δυνατότητα για parse και validate.*

Η χρήση της XML στην κωδικοποίηση των HL7 μηνυμάτων παρέχει τη δυνατότητα να γίνεται ανάγνωση των εισερχόμενων μηνυμάτων (parse) και αυθεντικοποίηση (validate) αυτών με οποιονδήποτε XML parser. Γι αυτό το σκοπό υπάρχουν πολλά έτοιμα προς χρήση εργαλεία (δωρεάν ή προς πώληση), όπως parsers, εφαρμογές που κάνουν μετασχηματισμούς, instance viewers και αρκετά άλλα, που κάνουν εύκολα το validation των μηνυμάτων ώστε οι εφαρμογές μας να μην επιφορτίζονται με αυτή τη λειτουργία.

- *Πιο εύκολη η χρήση της XML.*

Όσον αφορά στην κωδικοποίηση των μηνυμάτων είναι πολύ πιο εύκολο να χρησιμοποιηθεί η XML, αφού περισσότεροι είναι εξοικειωμένοι με τη χρήση της παρά με τη χρήση HL7 κώδικα. Αυτό βέβαια δε σημαίνει ότι η γνώση και κατανόηση του σημασιολογικού υπόβαθρου ενός HL7 μηνύματος δεν είναι αναγκαία.

- *Μείωση κόστους υλοποίησης.*

Μία HL7 εφαρμογή ενός οργανισμού υγείας περιλαμβάνει αρκετές υπό-εφαρμογές, όπως κάποιον parser και κάποιον generator (για τη δημιουργία των εξερχόμενων μηνυμάτων), που βελτιώνουν την επεξεργασία των παραδοσιακών HL7 μηνυμάτων. Αυτό όμως παρουσιάζει τεράστιο κόστος ανάπτυξης και υλοποίησης. Η μόνη εναλλακτική και αποδεκτή λύση ήταν η επιλογή μιας εκ των λίγων και ακριβών εμπορικών υλοποιήσεων για παραδοσιακά HL7 μηνύματα.

- *Συνεργασία των οργανισμών υγείας με άλλους φορείς.*

Με τη χρήση της XML γίνονται προσπελάσιμες πληροφορίες και δεδομένα ενός οργανισμού υγείας (π.χ. νοσοκομείο) από άλλους φορείς (π.χ. ασφαλιστικοί οργανισμοί) όπως και πληροφορίες από το Διαδίκτυο. Αυτό κάνει εφικτή την εισαγωγή χρήσιμης ή κρίσιμης ιατρικής πληροφορίας στα τοπικά συστήματα των μονάδων υγείας και των κλινικών τμημάτων, χωρίς ιδιαίτερη δυσκολία ή πολύπλοκες εγκαταστάσεις και προδιαγραφές.

- *Ομαλή μετάβαση σε νέες εκδόσεις του HL7.*

Η εδραίωση της χρήσης της XML σύνταξης για μηνύματα παλαιότερων εκδόσεων (2.x) βοηθά τους κατασκευαστές εφαρμογών και τους προμηθευτές να μεταβούν πιο ομαλά από την έκδοση 2 του



προτύπου στην έκδοση 3 ενθαρρύνοντας τη χρήση της XML στις εφαρμογές τους.

## **ΚΕΦΑΛΑΙΟ 4 : ΤΟ ΠΡΟΤΥΠΟ HL7**

### **4.1. ΓΕΝΙΚΑ**

*Ο* οργανισμός HL7 είναι ένας από τους πολλούς ANSI (American National Standards Institute) εγκεκριμένους οργανισμούς ανάπτυξης προτύπων [Standards Developing Organizations (SDOs)] και σχετίζεται με τον τομέα των υπηρεσιών στον χώρο της υγείας.

*Οι* περισσότεροι από αυτούς τους οργανισμούς παράγουν πρότυπα που έχουν σχέση με συγκεκριμένους τομείς της υγείας όπως είναι οι φαρμακευτικές και ιατρικές υπηρεσίες. Ο τομέας ενασχόλησης του HL7 είναι τα κλινικά και διαχειριστικά δεδομένα.

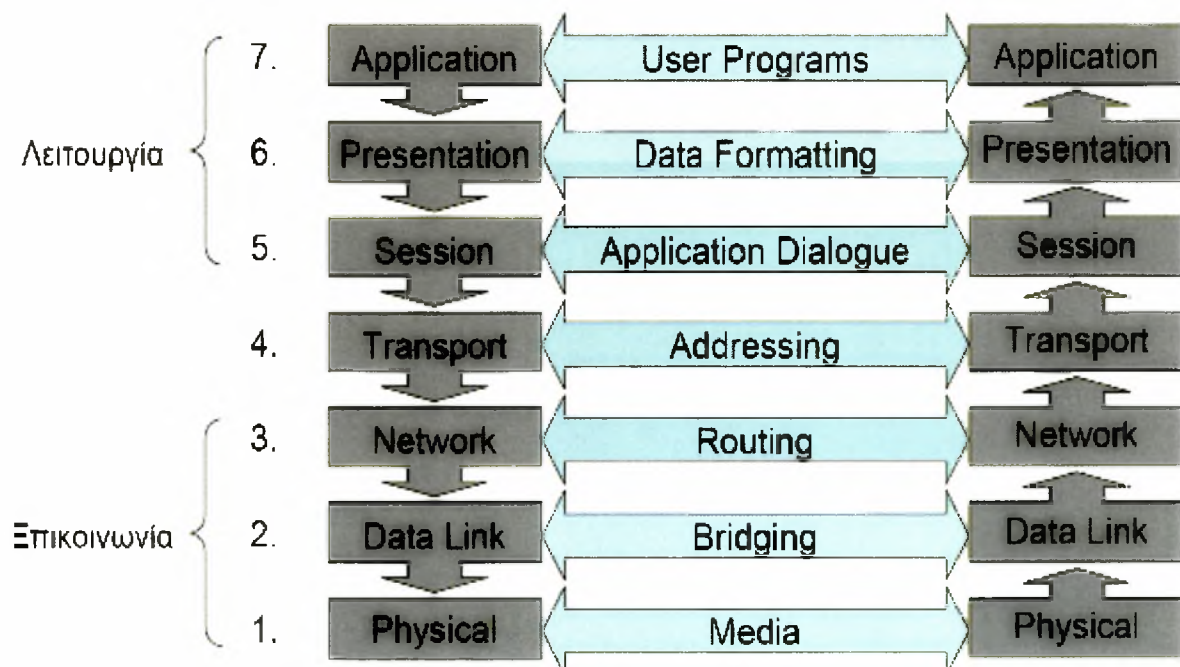
*Ο* οργανισμός HL7 λειτουργεί όπως οι περισσότεροι SDO οργανισμοί, είναι μη κερδοσκοπικός και εθελοντικός. Λειτουργεί με ένα αυστηρό και συγκεκριμένο τρόπο εκτέλεσης διεργασιών ώστε να διασφαλίζεται η συνέχεια, η διαφάνεια και η ισορροπία των συμφερόντων του. Στην πραγματικότητα ο HL7 αναπτύσσει πρότυπα που μπορούν να χρησιμοποιηθούν από διαφορετικές εφαρμογές, πάντα σε σχέση με την υγεία, για την ανταλλαγή κλινικών και διαχειριστικών δεδομένων.

### **4.2. ΤΙ ΣΗΜΑΙΝΕΙ Η ΟΝΟΜΑΣΙΑ HL7**

*Το* HL7 είναι ακρωνύμιο των λέξεων “Health Level 7”. Το 7 στο HL7, παραπέμπει στο υψηλότερο επίπεδο του μοντέλου επικοινωνίας για ανοικτά συστήματα του OSI (Open Systems Interconnection) του ISO (International Standards Organization), το επίπεδο εφαρμογής.

Το έβδομο επίπεδο (Level Seven) υποστηρίζει λειτουργίες όπως έλεγχος ασφάλειας, ταυτοποίηση εμπλεκόμενων τμημάτων στην επικοινωνία, έλεγχος διαθεσιμότητας, μηχανισμούς διαπραγματεύσεων ανταλλαγής και το πλέον σημαντικό, δομή ανταλλαγής δεδομένων.

Τα 7 επίπεδα του OSI



Ο Διεθνής Οργανισμός HL7 υφίσταται παγκόσμια και αναπτύσσει το πρότυπο ήδη από το 1987.

Η αποστολή του οργανισμού «HL7 Inc.» είναι η δημιουργία αξιόπιστων προτύπων ανταλλαγής, διαχείρισης και ολοκλήρωσης δεδομένων που αφορούν στην κλινική φροντίδα του ασθενούς και στην διαχείριση, οργάνωση και αξιολόγηση υπηρεσιών ιατρικής περίθαλψης. Ο οργανισμός ενθαρρύνει τη δημιουργία ευέλικτων

προτύπων, οδηγιών, μεθοδολογιών, πρωτοκόλλων και άλλων σχετικών υπηρεσιών και προϊόντων, προκειμένου να καταστεί εφικτή η διαλειτουργικότητα των πληροφοριακών συστημάτων στην Υγεία-Πρόνοια και η ανταλλαγή στοιχείων του ηλεκτρονικού φακέλου του ασθενούς.

*Ο οργανισμός "HL7 Inc."* δημιουργήθηκε προκειμένου να λειτουργεί ως αξιόπιστο μέσο επικοινωνίας μεταξύ των ενδιαφερόμενων φορέων στον τομέα της ιατρικής περίθαλψης. Το γεγονός αυτό αποτυπώνεται στην ποικιλία που παρουσιάζουν τα μέλη του. Αυτά μπορεί να είναι εταιρίες ιατρικής πληροφορικής, ιδιωτικοί και δημόσιοι φορείς υγείας και πρόνοιας, ειδικοί σύμβουλοι, εμπειρογνώμονες, εταιρίες ολοκληρωμένων πληροφοριακών συστημάτων (system integrators), ασφαλιστικοί φορείς, εταιρίες ιατροτεχνολογικού εξοπλισμού, φορείς παροχής υπηρεσιών υγείας-πρόνοιας, κλπ.

*Το HL7 περιλαμβάνει μία διεθνή κοινότητα που αποτελείται από εμπειρογνώμονες του χώρου της υγείας και επιστήμονες από το χώρο της πληροφορικής που συνεργάζονται με σκοπό να δημιουργηθούν πρότυπα για την ανταλλαγή, τη διαχείριση και την ολοκλήρωση των ηλεκτρονικών πληροφοριών υγειονομικής περίθαλψης. Το HL7 προωθεί τη χρήση τέτοιων προτύπων στο χώρο της υγείας για να αυξηθεί η αποτελεσματικότητα και η αποδοτικότητα των υπηρεσιών υγείας προς όφελος όλων.*

*Το HL7 είναι το πλέον ευρέως χρησιμοποιούμενο πρότυπο ανταλλαγής πληροφοριών μέσω μηνυμάτων σε κλινικό περιβάλλον. Χρησιμοποιείται σε όλες τις ηπείρους. Αν περιοριστεί κανείς στην Ευρώπη θα δει ότι χρησιμοποιείται σχεδόν σε κάθε χώρα ως πρότυπο ανταλλαγής πληροφοριών μέσω μηνυμάτων για διάφορα υποσυστήματα. Σχεδόν όλα τα ευφυή διαγνωστικά μηχανήματα (μέρος του ιατροτεχνολογικού εξοπλισμού) μπορούν να «συνομιλήσουν» μέσω του HL7. Τα περισσότερα ιατρικά*

πληροφοριακά συστήματα υψηλού επιπέδου είναι σε θέση να στείλουν και να λάβουν τα κατάλληλα HL7 μηνύματα, χρησιμοποιώντας τους κανόνες ανταλλαγής μηνυμάτων του πρωτοκόλλου αυτού. Το τρέχον πρότυπο βρίσκεται στην έκδοση 2.4 (2000), ενώ από το 1996 ο οργανισμός εργάζεται ήδη στην κατεύθυνση δημιουργίας μίας νέας γενιάς προτύπων βασισμένα στη γλώσσα XML (έκδοση 3).

Το πρωτόκολλο αυτό είναι ξεκάθαρα το πιο ώριμο πρότυπο ανταλλαγής πληροφοριών μέσω μηνυμάτων. Η έρευνα από την ακαδημαϊκή κοινότητα, τη βιομηχανία και τις εταιρίες συμβούλων οδήγησε σ' αυτό το πρότυπο, την κυριότητα του οποίου κατέχει ο μη κερδοσκοπικός οργανισμός Health Level Seven Inc. Το πρότυπο HL7 χρησιμοποιείται καθημερινά σε εκατοντάδες νοσοκομεία σε όλο τον κόσμο, συνδέοντας μια μεγάλη ποικιλία εφαρμογών και συστημάτων.

Αναγνωρίζοντας την ανάγκη υποστήριξης των τοπικών ομάδων που δραστηριοποιούνται στην προώθηση των προτύπων, ο "HL7 Inc." στηρίζει τις προσπάθειες αυτές με τη δημιουργία τοπικών παραρτημάτων (HL7 affiliates). Μέχρι σήμερα έχουν ήδη ιδρυθεί 23 τέτοια παραρτήματα (Ηνωμένο Βασίλειο, Ελλάδα, Καναδάς, Αυστραλία, Νέα Ζηλανδία, Νότιος Αφρική, Γερμανία, Ολλανδία, Φιλανδία, Ινδία, Ιαπωνία, Αργεντινή, Κίνα, Κορέα, Τσεχία, Λιθουανία, Ελβετία, Βραζιλία, Κροατία, Μεξικό, Ιταλία, Δανία και Ταϊβάν). Τα τοπικά παραρτήματα είναι ανεξάρτητοι οργανισμοί διεθνούς χαρακτήρα. Στοχεύουν στην ανάπτυξη, υποστήριξη, αποδοχή και χρήση των προτύπων του HL7 σε παγκόσμια κλίμακα με την μεταφορά τους στην αντίστοιχη γλώσσα του παραρτήματος.

Στο πλαίσιο αυτό, ο αποκλειστικός σκοπός του Ελληνικού οργανισμού είναι η προώθηση προτύπων ιατρικής πληροφορικής και ειδικότερα του προτύπου επικοινωνίας HL7 στα όρια της Ελληνικής Επικράτειας αλλά και στις χώρες της Νοτιοανατολικής Ευρώπης. Πιο

αναλυτικά, ο οργανισμός αυτός ασχολείται με τη διάδοση αξιόπιστων προτύπων ανταλλαγής, διαχείρισης και ολοκλήρωσης δεδομένων που αφορούν στην κλινική φροντίδα του ασθενούς και στην διαχείριση, οργάνωση και αξιολόγηση υπηρεσιών ιατρικής περίθαλψης. Επίσης, στόχος του HL7 Hellas είναι η δημιουργία ενός ηλεκτρονικού περιβάλλοντος, το οποίο θα λειτουργεί ως αξιόπιστο μέσο επικοινωνίας μεταξύ των ενδιαφερόμενων φορέων του τομέα της ιατρικής περίθαλψης. Επιπλέον, αρμοδιότητά του αποτελεί η συνεχής έρευνα και παρακολούθηση των εξελίξεων που σχετίζονται με τη βελτίωση του προτύπου HL7.

Στον οργανισμό αυτό συμμετέχουν κρατικά Ινστιτούτα, Πανεπιστήμια καθώς και ιδιωτικές εταιρείες. Συγκεκριμένα το Εργαστήριο Βιοϊατρικής Τεχνολογίας του Εθνικού Μετσόβιου Πολυτεχνείου (ΕΠΙΣΕΥ), το Ινστιτούτο Τεχνολογίας & Έρευνας, το Ινστιτούτο Ερευνών/Μελετών Τηλεπικοινωνιών και Πληροφορικής Χωρών Νοτιοανατολικής Ευρώπης (INA A.E.), το Εργαστήριο Πληροφορικής της Υγείας του Τμήματος Νοσηλευτικής του Πανεπιστημίου Αθηνών, η Κοινωνία Της Πληροφορίας Α.Ε., το Ινστιτούτο Βιοϊατρικής Τεχνολογίας, το Εργαστήριο Ιατρικής Φυσικής του Δημοκρίτειου Πανεπιστημίου Θράκης, η DATAMED A. E., η APOLLO A. E., η Computer Control Systems A.E., η Φ. Πασχίδου & ΣΙΑ Ε.Ε., η ΓΝΩΜΩΝ Πληροφορικής Α.Ε., η EXODUS A.E., η SAP Hellas A.E. και η 01 Πληροφορική Α.Ε.

#### **4.3. ΓΕΝΙΚΗ ΠΕΡΙΓΡΑΦΗ ΤΟΥ ΠΡΩΤΟΚΟΛΛΟΥ HL7**

Χωρίς να χρησιμοποιηθούν ειδικοί τεχνικοί όροι, το πρότυπο μπορεί να οριστεί σαν ένα εργαλείο, με ειδική εγγραφή σε κάθε σύστημα, που διευκολύνει διαφορετικά συστήματα ή ιατρικά μηχανήματα, να επικοινωνούν με ευχέρεια μεταξύ τους και να διαβιβάζουν τα

δεδομένα που περιέχουν το ένα στο άλλο. Με τη χρήση του προτύπου, για παράδειγμα, ο βιοχημικός αναλυτής, στο εργαστήριο νοσοκομείου, μπορεί να δέχεται μέσω ηλεκτρονικού υπολογιστή απευθείας εντολές εξετάσεων από τα κλινικά τμήματα και να επιστρέφει αυτόματα τις απαντήσεις αυτών, στα τμήματα που τις παρήγγειλαν. Η επικοινωνία αυτή, μεταξύ του αναλυτή και του υπολογιστή της κλινικής, τεχνικά ονομάζεται διαλειτουργικότητα. Το πρότυπο HL7 είναι λοιπόν ένας κώδικας επικοινωνίας, κοινά αποδεκτός από όλους τους κατασκευαστές. Στο βαθμό που θα εγκατασταθεί και θα λειτουργήσει, προσφέρει ουσιαστική βοήθεια στον γιατρό, γιατί τον απαλλάσσει από τον φόρτο της “χειρωνακτικής” διαχείρισης τεράστιου όγκου ιατρικής πληροφορίας, που του απορροφά σημαντικό χρόνο και τον απομακρύνει από τον διαγνωστικό και θεραπευτικό του στόχο.

Το πρότυπο HL7 δεν αφορά αποκλειστικά στη διαβίβαση πληροφορίας μεταξύ εργαστηρίου και κλινικής. Έχει δομηθεί έτσι ώστε να εμπεριέχει, εκτός από κλινικά και εργαστηριακά δεδομένα και κάθε άλλη υπαρκτή πληροφορία σε οποιοδήποτε νοσηλευτικό ίδρυμα και υγειονομικό σύστημα. Στο ίδιο τμήμα του HL7 μηνύματος εμπεριέχονται ασφαλιστικά και οικονομικά στοιχεία, προμήθειες και διαχείριση υλικού, φάρμακα και εργασίες, αναλώσιμος και πάγιος εξοπλισμός. Βασική προϋπόθεση είναι, τα συστήματα να διασυνδεθούν και το καθένα να μπορεί να έχει τα απαραίτητα στοιχεία, ώστε να διεκπεραιώνει αυτόματα όλες τις προσφερόμενες λειτουργίες συν της βασικής (κλινική λειτουργία). Αποφεύγεται έτσι εντελώς η γραφειοκρατία, εφόσον μία και μοναδική εγγραφή, για κάθε ασθενή, μπορεί να διανέμεται εύκολα, σε κάθε κλινικό, εργαστηριακό ή διοικητικό τμήμα, ανάλογα με τις ανάγκες του καθενός. Για το λόγο αυτό, το πρότυπο HL7 επικρατεί, εφόσον διατηρεί ένα συντριπτικό πλεονέκτημα. Αντιμετωπίζει ένα νοσοκομείο ή ένα ευρύτερο σύστημα υγείας, σαν ενιαία λειτουργική μονάδα, όπως πράγματι είναι.



#### **4.4. ΠΕΡΙΓΡΑΦΗ ΤΗΣ ΑΡΧΙΤΕΚΤΟΝΙΚΗΣ ΤΟΥ HL7**

##### **4.4.1. ΕΙΣΑΓΩΓΗ**

*Το HL7 είναι ένα δομημένο (structured), μηνυματοκεντρικό (message-oriented) πρωτόκολλο που εξασφαλίζει την επικοινωνία υπολογιστών και υπηρεσιών ιατρικής περίθαλψης. Αρχικά το HL7 είχε σχεδιαστεί ως ένα πρωτόκολλο του επιπέδου επτά του OSI αλλά με την πάροδο των ετών αναγνωρίστηκε ως εγκεκριμένο πρωτόκολλο από τον ANSI Αμερικής και τον EDI Γερμανίας.*

*Η αρχιτεκτονική του πρωτοκόλλου είναι ιεραρχική. Κάτι που σημαίνει ότι ξεκινά από υψηλό επίπεδο ομαδοποίησης και δόμησης των δεδομένων και καταλήγει σε μία πληθώρα πεδίων τιμών. Κάθε επίπεδο της ιεραρχίας εξυπηρετεί συγκεκριμένη λειτουργία.*

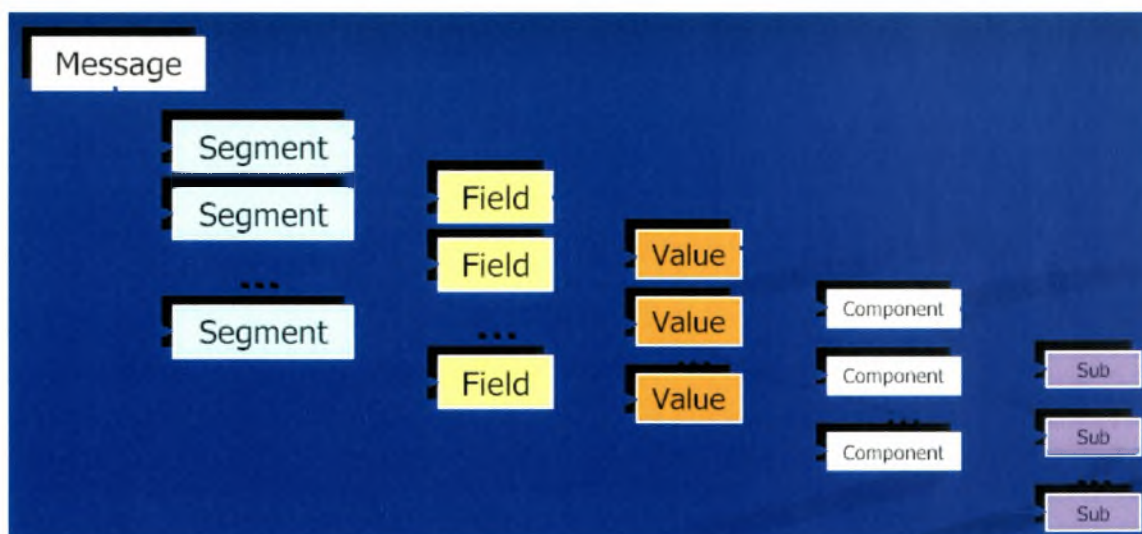
*Όσον αφορά στη δομή (ακολουθεί ενδεικτικό σχεδιάγραμμα), πρέπει να αναφέρουμε ότι πολλά πεδία (fields) μαζί, αποτελούμενα από διάφορους τύπους δεδομένων (data types), συνιστούν ένα κομμάτι (segment). Το σύνολο των κομματιών αποτελεί μια περιγραφή μηνύματος (message definition) και με τη σειρά τους τα μηνύματα συνιστούν τις λειτουργικές ομάδες (functional groups).*

*Όσον αφορά στη λειτουργικότητα του πρωτοκόλλου, το HL7 μπορεί να θεωρηθεί ότι είναι ένα σύνολο από πρωτόκολλα. Κάθε μήνυμα ανάλογα με τον ορισμό του κατατάσσεται σε μία λειτουργική ομάδα. Αυτό έχει ως αποτέλεσμα τη διαφοροποίηση του τρόπου δόμησης των μηνυμάτων και της περιγραφής των λειτουργικών ομάδων.*



Το HL7 δεν είναι βασισμένο σε κάποιο ήδη υπάρχον αυθαίρετο μοντέλο δεδομένων, άρα δεν έχει δοκιμαστεί σε κάποια προϋπάρχουσα αρχιτεκτονική. Η χρήση ενός ήδη εφαρμοσμένου μοντέλου δεδομένων αν και σίγουρα θα έδινε αρκετά πλεονεκτήματα θα μείωνε την ουσιαστική χρησιμότητα του πρωτοκόλλου κι αυτό γιατί το πρότυπο εφαρμόζεται σε ετερογενή συστήματα τα οποία παρουσιάζουν αργή εξέλιξη.

#### Δομή του HL7 μηνύματος



#### **4.4.2. FIELDS – ΠΕΔΙΑ**

Στο πεδίο μπορούν να αποθηκευτούν εκατοντάδες πληροφορίες σχετικές με έναν ασθενή αλλά και στοιχεία που αφορούν κλινικές και οικονομικές πληροφορίες. Το HL7 χρησιμοποιεί περισσότερους από 10 αφηρημένους (abstract) τύπους δεδομένων, (data types) για

να καθορίσει την φύση των πεδίων. Σαν συνέπεια, κάποια πεδία έχουν περισσότερα από ένα στοιχεία.

Για παράδειγμα ένα πεδίο που έχει χρονοσφραγίδα (time stamp TS) ακολουθεί ένα προκαθορισμένο τύπο. Επίσης πολλά πεδία είναι (ή είναι εύκολο να μετατραπούν) κωδικοποιημένα. Το πρότυπο περιέχει πολλούς πίνακες κωδικοποίησης που καθορίζουν αποδεκτά περιεχόμενα. Κάθε πεδίο έχει καθορισμένο μέγιστο μήκος.

#### **4.4.3. SEGMENT DEFINITION – ΚΑΘΟΡΙΣΜΟΣ ΚΟΜΜΑΤΙΟΥ**

Τα κομμάτια (segments) παρέχουν μια λογική ομαδοποίηση της πληροφορίας. Ο τρόπος ομαδοποίησης των πεδίων σε κομμάτια είναι μέρος του προτεινόμενου μοντέλου πληροφορίας του HL7. Τα κομμάτια μπορεί να είναι υποχρεωτικά ή προαιρετικά, μπορεί να είναι εμφωλιασμένα (nested) ή μπορεί να επαναλαμβάνονται. Ένα μήνυμα μπορεί να πάρει αυθαίρετη αλλά πάντα κατανοητή και ευανάγνωστη μορφή.

#### **4.4.4. MESSAGE DEFINITION – ΠΕΡΙΓΡΑΦΗ ΜΗΝΥΜΑΤΟΣ**

Μέσα σε κάθε τύπο μηνύματος υπάρχουν μία ή περισσότερες περιγραφές μηνυμάτων, που καθορίζουν το ακριβές σύνολο ή το συνδυασμό κομματιών (segments) που ορίζουν ένα μήνυμα. Για παράδειγμα, το ADT μπορεί να υπάρξει σε περισσότερες από 30 διαφορετικές περιγραφές μηνυμάτων των «γεγονότων ώθησης-σκανδάλης» (trigger events) ή άλλων κανόνων. Κάθε καθορισμός μηνύματος εμπεριέχει ένα ή περισσότερα κομμάτια.

#### **4.4.5. MESSAGE TYPE – ΤΥΠΟΣ ΜΗΝΥΜΑΤΟΣ**

Μέσα σε μια λειτουργική ομάδα περιγράφονται ένας ή περισσότεροι τύποι μηνυμάτων που μπορούν να υλοποιηθούν σε διάφορους συνδυασμούς για να υποστηρίξουν υψηλού επιπέδου επιχειρησιακούς κανόνες(business rules). Για παράδειγμα, το ADT συγκεκριμενοποιεί ένα μήνυμα ενώ το Order Entry περιγράφει περισσότερα από δώδεκα.

#### **4.4.6. FUNCTIONAL GROUP – ΛΕΙΤΟΥΡΓΙΚΗ ΟΜΑΔΑ**

Περιοχές του πρωτοκόλλου ομαδοποιούνται ανάλογα με τις κοινές λειτουργικές τους εφαρμογές. Για παράδειγμα ADT, Order Entry, Finance, Control και Ancillary Reporting όλα αντιπροσωπεύουν ομάδες που περιγράφονται στο πρότυπο. Οι διάφορες λειτουργικές ομάδες δίνονται τυπικά στα αντίστοιχα κεφάλαια του προτύπου HL7.

#### **4.4.7. ΠΑΡΑΔΕΙΓΜΑ ΜΗΝΥΜΑΤΟΣ HL7**

```
MSH|^~\&|LAB|767543|ADT|767543|19900314130405||ADT^A04|XX
3657|P|2.3.1<CR>
EVN|A01|19980327101314|19980327095000|I||19980327095000<CR
>
PID|1||123456789ABCDEF|123456789ABCDEF|PATIENT^BOB^S||1
9590520|M||
612345 MAIN STREET^^ANYTOWN^CA^91234||714-555-1212|
714-555-1212|||123456789ABCDEF|||U<CR>
PDI|||WELBY<CR>
PVI|1|0||NEW||SPOCK<CR>
```

Στο παράδειγμα αυτό αποστέλλεται ένα μήνυμα, το οποίο είναι ADT\_A04 για την εισαγωγή ενός ασθενούς και φαίνονται μόνο τα δεδομένα που αποστέλλονται. Στα χρωματισμένα στοιχεία, βλέπουμε την αρχή των segments όπου το πρώτο στοιχείο που αποστέλλεται είναι το όνομα του κομματιού (segment), στο συγκεκριμένο

παράδειγμα: MSH, EVN, PID, PD1, PV1. Μέσα στο κομμάτι υπάρχουν διάφορα ομαδοποιημένα δεδομένα και μέσα στις ομάδες δεδομένων κάποιες υποομάδες. Βλέπουμε στο μήνυμα δηλαδή να διαχωρίζονται τα segments και μέσα στα segments οι πρώτες ομάδες δεδομένων (fields) να διαχωρίζονται με το χαρακτήρα ‘|’(field separator). Αν τα fields συμπεριλαμβάνουν περισσότερους από έναν τύπους δεδομένων αυτό φαίνεται με ακόμα ένα χαρακτήρα κωδικοποίησης ‘^’. Φαίνεται δηλαδή η ιεραρχική δομή μέσα στο μήνυμα. Πολλά μηνύματα μαζί ομαδοποιούνται σε μια λειτουργική ομάδα και μέσα σε αυτή μπορούν να υπάρξουν επιμέρους ομαδοποιήσεις. Ένα διαφορετικό σχήμα για την δομή είναι το πιο κάτω:

### **Λειτουργική Ομάδα**



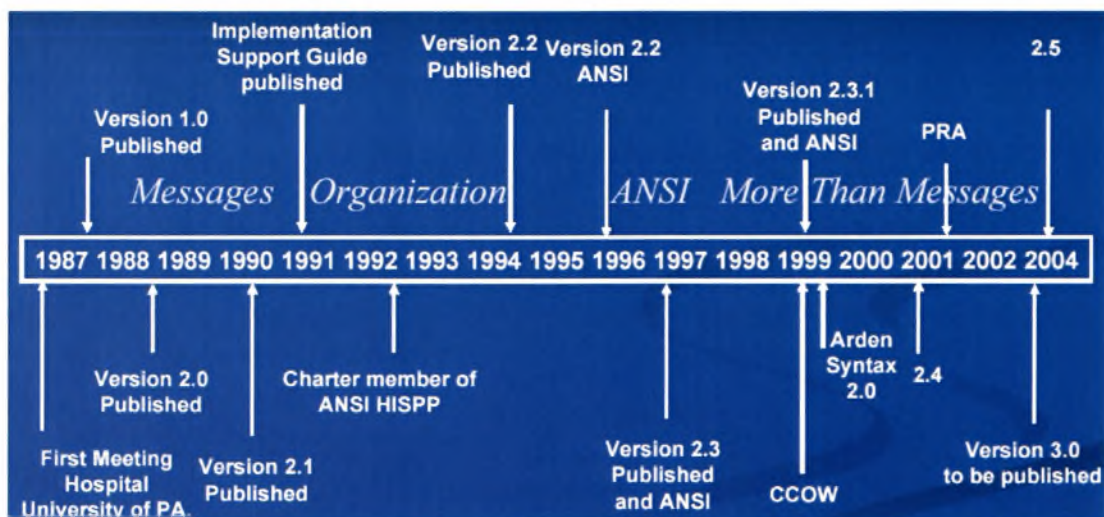
## ΚΕΦΑΛΑΙΟ 5 : ΧΑΡΑΚΤΗΡΙΣΤΙΚΑ ΤΟΥ HL7

### 5.1. ΕΙΣΑΓΩΓΗ

Σε προηγούμενο κεφάλαιο περιγράψαμε το πρωτόκολλο HL 7 χωρίς πολλούς τεχνικούς όρους. Στη συνέχεια θα παρουσιάσουμε τον σκοπό του πρωτοκόλλου, από πού προήλθε η ιδέα ανάπτυξής του και πού βασίζεται αυτή, την ανάγκη για δημιουργία ενός προτύπου, τους στόχους που προσπαθεί το πρότυπο να επιτύχει και επίσης την ιστορία της εξέλιξης του προτύπου. Ακόμα, αναλυτικά δίνεται μια συνολική περιγραφή των βασικών στοιχείων αυτού, των προοπτικών εξέλιξης του πρωτοκόλλου και της σχέσης του με άλλα πρωτόκολλα. Στη συνέχεια παραθέτουμε δείγμα των δυνατοτήτων του πρωτοκόλλου που αφορούν στην ασφάλεια του, στην παρουσία του ως μια ολοκληρωμένη λύση, στη διαλειτουργικότητα του ως ένα από τα πιο σημαντικά στοιχεία που έχει, καθώς και στη δυνατότητά του να επεξεργαστεί στοιχεία από παλαιότερα συστήματα καταγραφής και διατήρησης ιατρικών δεδομένων, είτε αυτά είναι ηλεκτρονικά είτε όχι.

Ο σκοπός και η ιδέα της δημιουργίας ενός πρωτοκόλλου επικοινωνίας μεταξύ ιατρικών μηχανημάτων γεννήθηκε το 1987 στο πανεπιστήμιο της Πενσυλβάνιας. Εκεί αναγνωρίστηκε η ανάγκη απλοποίησης της υλοποίησης των διεπαφών των εφαρμογών των ηλεκτρονικών υπολογιστών ανάμεσα σε διάφορους ανταγωνιστικούς κατασκευαστές. Έτσι δημιουργήθηκε μια επιτροπή που αποτελείτο από χρήστες και κατασκευαστές. Αυτή η επιτροπή αργότερα ονομάστηκε Ομάδα Εργασίας του HL7 (HL7 Working Group). Ομάδες από άλλες χώρες υπάρχουν και δουλεύουν παράλληλα. Όλη αυτή η προσπάθεια έχει σαν σκοπό τη δημιουργία ενός προτύπου που θα είναι ανοικτό σε όλους όσους αναπτύσσουν συστήματα επεξεργασίας της ιατρικής πληροφορίας.

Στο πιο κάτω σχεδιάγραμμα φαίνεται συνοπτικά η ιστορία του HL7:



## 5.2. ΥΠΟΒΑΘΡΟ

Ο αριθμός 7 στο HL7 αναφέρεται στο έβδομο επίπεδο του OSI (Open System Interconnection) μοντέλου. Πρόκειται για έναν όρο που υπονοεί ότι αφορά σε ένα μοντέλο για επικοινωνία ανάμεσα σε εφαρμογές, που γίνεται στο έβδομο επίπεδο του OSI. Οι εφαρμογές αυτές σχετίζονται με τον ορισμό των δεδομένων για ανταλλαγή και με τον συγχρονισμό των εναλλασσόμενων στοιχείων. Ωστόσο, η ανάγκη κατανόησης του πρωτοκόλλου από τους χρήστες του, επιβάλλει την ύπαρξη κάποιων σημείων που αναφέρονται σε πιο χαμηλά επίπεδα. Κάποιες άλλες φορές αυτές οι αναφορές είναι απαραίτητες για να μπορούν να δημιουργηθούν συστήματα που βασίζονται στο HL7. Το πρότυπο δεν είναι φτιαγμένο για να δουλεύει με μια συγκεκριμένη αρχιτεκτονική. Έχει σχεδιαστεί προκειμένου να υποστηρίζει κεντρικά συστήματα ιατρικής περίθαλψης αλλά και περισσότερο κατανεμημένα περιβάλλοντα. Πρόκειται λοιπόν για ένα πρωτόκολλο του οποίου κύριο



χαρακτηριστικό είναι η επίτευξη της διαλειτουργικότητας, της επικοινωνίας δηλαδή διαφορετικών εφαρμογών ανάμεσα από πολλαπλά ετερογενή συστήματα.

### **5.3. Η ΑΝΑΓΚΗ ΓΙΑ ΠΡΟΤΥΠΟ**

Το σύνολο του ιατροτεχνολογικού εξοπλισμού σε αρκετές χώρες είναι ήδη αρκετά μεγάλο και συνεχώς αυξάνεται. Παρά ταύτα ο υπάρχον εξοπλισμός δεν λειτουργεί ενιαία, διότι δεν αξιοποιείται η δυνατότητα για επικοινωνία και ανταλλαγή, κάτι που έχει ως αποτέλεσμα απώλειες ιατρικής και διαχειριστικής πληροφορίας. Η υπολογιστική και τεχνολογική ισχύς στις μονάδες και στις υπηρεσίες υγείας, χωρίς την εκμετάλλευση της διαλειτουργικότητας και της διασύνδεσης, με ένα και μόνο πρότυπο, καταλήγει στην δημιουργία κοινωνικού και οικονομικού φόρτου, ιδιαίτερα επικίνδυνου για το άμεσο μέλλον κάθε χώρας. Προκειμένου να αποφευχθεί το παραπάνω, εναλλακτική λύση ουσίας αποτελεί η λειτουργική διασύνδεση του ιατροτεχνολογικού εξοπλισμού, που ήδη υπάρχει σε λειτουργία ή παραμένει αχρησιμοποίητος στις αποθήκες των νοσοκομείων και των ασφαλιστικών οργανισμών.

Στις περασμένες δύο δεκαετίες τα ιδρύματα υγείας άρχισαν να αυτοματοποιούν διάφορες πλευρές της διαχείρισης πληροφορίας. Στην αρχή οι προσπάθειες γίνονταν για να περιοριστεί η γραφειοκρατία και να γίνει καλύτερη διαχείριση των χρηματικών πόρων. Στη συνέχεια οι προσπάθειες αυτές στράφηκαν προς τη βελτίωση των ιατρικών υπηρεσιών που παρέχονταν στους ασθενείς. Τα τελευταία χρόνια το ενδιαφέρον έχει εστιαστεί στο να συλλέγονται και να οργανώνονται όλες οι ιατρικές πληροφορίες, κάτι δηλαδή σαν ηλεκτρονικός φάκελος του ασθενούς. Είναι σύνηθες φαινόμενο για ένα μέσο νοσοκομείο να έχει διαφορετικά υπολογιστικά συστήματα για την εισαγωγή του ασθενούς, για τις

ιατρικές εξετάσεις, για τις ακτινογραφίες κτλ. Είναι λοιπόν εμφανής η ανάγκη για ένα μέσο που θα μπορούσε να μεταφέρει αυτές τις διαφορετικές ομάδες πληροφορίας από ένα σύστημα σε ένα άλλο, απαλλάσσοντας έτσι το ίδρυμα από περιττά έξοδα και χρόνο.

#### **5.4. ΣΤΟΧΟΙ ΤΟΥ ΠΡΟΤΥΠΟΥ**

Το πρότυπο HL7 δημιουργήθηκε με σκοπό την εκπλήρωση κάποιων «a priori» στόχων. Κύριος στόχος του προτύπου είναι να παρέχει στάνταρτ για την ανταλλαγή δεδομένων μεταξύ των υπολογιστών ιατρικών μηχανημάτων. Τα στάνταρτ αυτά μειώνουν την υλοποίηση ειδικών προγραμμάτων σε κάθε μηχανήμα, κάτι που ήταν απαραίτητο προκειμένου να επιτευχθεί η επικοινωνία κι ανταλλαγή. Ένας ακόμη στόχος όμως του προτύπου είναι και η υποστήριξη τέτοιων προγραμμάτων.

Πιο αναλυτικά η αρχική διατύπωση των στόχων που τέθηκαν ήταν η εξής:

- Το πρότυπο θα πρέπει να υποστηρίζει την ανταλλαγή πληροφοριών ανάμεσα σε συστήματα που ανήκουν σε ένα ευρύτερο φάσμα τεχνικών χαρακτηριστικών και αρχιτεκτονικών. Θα πρέπει να μπορεί να εφαρμοστεί και πρακτικά σε συστήματα που υποστηρίζουν ποικιλία προγραμματιστικών γλωσσών, λειτουργικών και τηλεπικοινωνιακών συστημάτων.
- Πρέπει να υποστηρίζεται τόσο η άμεση αποστολή αυτόνομων συναλλαγών, όσο και η μεταφορά πολλαπλών συναλλαγών με τη χρήση αρχείων.
- Το πρότυπο πρέπει να εξελίσσεται και να μεγαλώνει προκειμένου να αντιμετωπίζει νέες απαιτήσεις που τυχόν προκύπτουν. Στα πλαίσια της εξέλιξης αυτής θα πρέπει να



προστίθενται συνεχώς νέα κομμάτια και καινούριες υλοποιήσεις σε ήδη υπάρχοντα περιβάλλοντα.

- Αν και πρώτος στόχος είναι το πρότυπο να επικεντρωθεί σε νοσοκομειακά συστήματα πληροφορίας, ένας μακροπρόθεσμος είναι να αποτελέσει πρότυπο για όλα τα ιατρικά πληροφοριακά συστήματα οποιασδήποτε υπηρεσίας.
- Η διαφορετική φύση των επιχειρηματικών δραστηριοτήτων που αναπτύσσονται γύρω από ένα φορέα υγείας εμποδίζει την ύπαρξη ενός πρωτοκόλλου που θα είναι γενικό και θα μπορεί να χρησιμοποιηθεί παντού και αμέσως.
- Το HL7 δεν κάνει a priori υποθέσεις για την αρχιτεκτονική των εφαρμογών, ούτε προσπαθεί να ομαλοποιήσει τις διαφοροποιήσεις που παρουσιάζονται στην αρχιτεκτονική των υπολογιστικών συστημάτων. Οι διαφορές αυτές πρέπει να λύνονται επί τόπου όταν εφαρμοστεί το HL7.
- Ένας από τους κύριους στόχους του HL7 είναι να χρησιμοποιηθεί το πρότυπο το συντομότερο δυνατό. Επειδή αυτό έχει γίνει ήδη, ο HL7 έχει μια υποδομή που υποστηρίζει το πρότυπο και έχει αναγνωριστεί από το American National Standards Institute (ANSI) σαν ένα Accredited Standards Organization (ASO).
- Η συνεργασία με άλλα σχετικά προγράμματα προτυποποίησης ιατρικών υπηρεσιών είναι κύριο μέλημα του οργανισμού. Ο HL7 συνεργάζεται με τα ACR/NEMA DICOM, ASC X12, ASTM, IEEE/MEDIX, NCPDP και άλλα.

### **5.5. ΣΤΡΑΤΗΓΙΚΕΣ ΕΔΡΑΙΩΣΗΣ ΤΟΥ HL7**

Προκειμένου να εδραιωθεί το πρωτόκολλο σε παγκόσμιο επίπεδο, ο οργανισμός HL7 ακολουθεί συγκεκριμένες πολιτικές εδραίωσής του:

1. Ανάπτυξη συνεπών και με δυνατότητα επέκτασης προτύπων, που θα επιτρέπουν τη χρήση δομημένων και κωδικοποιημένων πληροφοριών υγειονομικής περίθαλψης. Έτσι θα είναι δυνατό να επιτρέπεται η ανταλλαγή δεδομένων μεταξύ υπολογιστικών εφαρμογών, χωρίς να αλλοιώνεται η σημασία τους.
2. Ανάπτυξη μιας επίσημης μεθοδολογίας για την υποστήριξη της κατασκευής των HL7 προτύπων από το HL7 Μοντέλο Αναφοράς Πληροφοριών: HL7 Reference Information Model (RIM).
3. Ενημέρωση της βιομηχανίας υγειονομικής περίθαλψης, των πολιτικών και πολιτειακών φορέων και του ευρύτερου κοινού γενικότερα σχετικά με τα οφέλη της τυποποίησης πληροφοριών υγειονομικής περίθαλψης και το HL7 πρότυπο.
4. Προώθηση της χρήσης του HL7 παγκοσμίως μέσω της δημιουργίας HL7 διεθνών θυγατρικών οργανισμών. Οι οργανισμοί αυτοί συμμετέχουν στην ανάπτυξη των HL7 προτύπων και στην τροποποίηση τους ανάλογα με το υγειονομικό σύστημα και τις ανάγκες αυτού σε κάθε χώρα.
5. Ενθάρρυνση επιστημόνων του τομέα της Πληροφορικής για συμμετοχή τους στην ανάπτυξη και εδραίωση του HL7.
6. Συνεργασία και με λοιπούς οργανισμούς ανάπτυξης προτύπων καθώς και με οργανισμούς έγκρισης και επικύρωσης αυτών (π.χ. Ansi και ISO).
7. Συνεργασία με τους χρήστες πληροφοριών υγειονομικής περίθαλψης, ώστε να εξασφαλισθεί ότι τα HL7 πρότυπα καλύπτουν πραγματικές και πιθανές μελλοντικές ανάγκες.

## **5.6. ΛΟΓΟΙ ΧΡΗΣΗΣ ΤΟΥ HL7**

Αυτή την περίοδο, σε όλο τον κόσμο εξελίσσονται πολλές προσπάθειες ανάπτυξης προτύπων υγειονομικής περίθαλψης. Προκύπτει λοιπόν το εύλογο ερώτημα: Γιατί τόσες προσπάθειες εδραίωσης του HL7. Υπάρχουν αρκετοί λόγοι για τους οποίους προωθείται τόσο το συγκεκριμένο πρότυπο και μερικοί από αυτούς παρουσιάζονται στη συνέχεια.

- Το HL7 είναι καινοτόμο πρωτόκολλο αφού εστιάζει στις απαιτήσεις των διεπαφών όλων των οργανισμών-επιχειρήσεων υγειονομικής περίθαλψης, ενώ οι υπόλοιπες προσπάθειες ανάπτυξης τέτοιων προτύπων εστιάζουν στις απαιτήσεις ενός συγκεκριμένου τομέα.
- Η ομάδα ανάπτυξης του HL7 εξετάζει τις απαιτήσεις του ήδη εγκατεστημένου πρωτοκόλλου σε νοσοκομεία που χρησιμοποιούν νέες τεχνολογίες.
- Ενώ το HL7 εστιάζει στην εξέταση και επίλυση άμεσων αναγκών, η ομάδα ανάπτυξής του συνεχίζει να αφιερώνει τις προσπάθειές της στην εξασφάλιση συμφωνίας με τις λοιπές Ηνωμένες Πολιτείες και με διεθνείς οργανισμούς ανάπτυξης προτύπων.
- Η Αργεντινή, η Αυστραλία, ο Καναδάς, η Κίνα, η Τσεχία, η Φινλανδία, η Γερμανία, η Ινδία, η Ιαπωνία, η Κορέα, η Λιθουανία, οι Κάτω Χώρες, η Νέα Ζηλανδία, η νότια Αφρική, η Ελβετία, η Ταϊβάν, η Τουρκία και το Ηνωμένο Βασίλειο είναι μέλη των HL7 πρωτοβουλιών.
- Επιπλέον, το HL7 είναι εγκεκριμένο πρότυπο από τον ANSI.
- Το HL7 προσπαθεί να προσδιορίσει και να υποστηρίξει όλες τις διαφορετικές απαιτήσεις κάθε εμπλεκόμενου φορέα είτε είναι χρήστης είτε προμηθευτής είτε σύμβουλος.
- Χρησιμοποιείται σε παγκόσμια κλίμακα (υπάρχουν 30 εθνικά παραρτήματα).
- Χρησιμοποιείται στην Ευρωπαϊκή Ένωση (HL7/TC251 task force).

- Χρησιμοποιείται παραγωγικά παραπάνω από 10 χρόνια (τρέχουσα έκδοση 2.5).
- Έχει την υποστήριξη της διεθνούς αγοράς, αφού οι κατασκευαστές λογισμικού διαθέτουν την απαιτούμενη εμπειρία και υπάρχει σαφής εικόνα για τις ανάγκες των χρηστών.
- Πρόκειται για ΑΝΟΙΧΤΟ πρότυπο που ανανεώνεται διαρκώς (δράσεις έκδοσης 3 , χρήση XML).
- Το απαγορευτικό κόστος ανάπτυξης τέτοιων προτύπων στο παρελθόν, έχει αρχίσει να μειώνεται ραγδαία, με συνέπεια η επαφή του γιατρού με τα νέα τεχνολογικά συστήματα να είναι απλούστερη.
- Το τεχνικό πρόβλημα της διαλειτουργικότητας, μεταξύ των πληροφοριακών συστημάτων υγείας, φαίνεται να λύνεται ικανοποιητικά με το πρότυπο HL7, κάτι που δίνει νέα ώθηση στην τροποποίηση και στη βελτίωση της ιατρικής.

## **5.7. ΝΕΕΣ ΚΑΙ ΤΡΕΧΟΥΣΕΣ ΠΡΩΤΟΒΟΥΛΙΕΣ**

### **5.7.1. HIPAA**

*Η* αρχική συμμετοχή του HL7 στην πρωτοβουλία HIPAA ξεκίνησε το 1996 με την συγκρότηση της ομάδας Claims Attachments. Αυτή η ομάδα συγκροτήθηκε προκειμένου να τυποποιήσει συμπληρωματικές πληροφορίες που απαιτούντο για την ασφάλεια της υγειονομικής περίθαλψης καθώς και άλλων συναλλαγών ηλεκτρονικού εμπορίου. Τα μελλοντικά προγράμματα του Claims Attachments περιλαμβάνουν την Υγεία στο Σπίτι, την Ειδικευμένη Δυνατότητα Νοσηλείας, τον Ανθεκτικό Ιατρικό Εξοπλισμό (DME) και την αυθεντικοποίηση των χρηστών. Δεν περιορίζεται όμως μόνο σε αυτά. Μια σημαντική προσπάθεια, έπειτα από καθοδήγηση από την ομάδα ανάπτυξης του HL7, γίνεται και στον τομέα της ενίσχυσης των διοικητικών εφαρμογών. Περισσότερες πληροφορίες για τα HIPAA πρότυπα μπορούν να βρεθούν στην ιστοσελίδα του Washington Publishing.

### **5.7.2. ΑΝΑΦΟΡΙΚΟ ΜΟΝΤΕΛΟ ΠΛΗΡΟΦΟΡΙΩΝ - Reference Information Model (RIM)**

Το αναφορικό μοντέλο πληροφοριών (RIM) είναι ο ακρογωνιαίος λίθος της έκδοσης 3 του HL7. Το RIM, με απλά λόγια, είναι μια μεγάλη εικονογραφική αντιπροσώπευση των κλινικών δεδομένων (domains). Προσδιορίζει τον κύκλο ζωής των γεγονότων που θα φέρει ένα μήνυμα ή ομάδες μηνυμάτων. Είναι ένα κοινό πρότυπο μεταξύ όλων των domains για τη δημιουργία μηνυμάτων. Επίσης, αντιπροσωπεύει τις συνδέσεις που υπάρχουν μεταξύ των πληροφοριών των πεδίων (fields) του HL7 μηνύματος, κάτι που συνεπάγεται την αύξηση της ακριβείας και τη μείωση των δαπανών ανάπτυξης.

### **5.7.3. ΛΕΞΙΛΟΓΙΟ**

Το πρότυπο HL7 προσφέρει τη δυνατότητα ανταλλαγής μηνυμάτων μεταξύ συστημάτων. Η χρησιμότητα όμως αυτής της δυνατότητας μπορεί να υποβιβαστεί, εκτός κι αν από κοινού υπάρχει καλά καθορισμένη γνώση της σημασίας των δεδομένων που μεταφέρονται. Δεδομένου ότι ένα μεγάλο μέρος της μεταφερόμενης πληροφορίας κωδικοποιείται, είτε από το HL7 είτε από άλλους οργανισμούς, το HL7 ξεκίνησε μια προσπάθεια οργάνωσης της ορολογίας των HL7 μηνυμάτων μέσω του σχηματισμού της Τεχνικής Επιτροπής Λεξιλογίου (Vocabulary Technical Committee).

Αυτή η επιτροπή κάνει σημαντικές προσπάθειες προκειμένου να οργανώσει και να αποθηκεύσει το κωδικοποιημένο λεξιλόγιο. Το λεξιλόγιο αυτό χρησιμοποιείται από το HL7 και από όλα τα σχετικά πρότυπα και επιτρέπει την ανταλλαγή κλινικών στοιχείων και πληροφοριών. Με αυτό τον τρόπο όλα τα εμπλεκόμενα συστήματα σε μία ανταλλαγή δεδομένων έχουν καθορισμένη γνώση της σημασίας των στοιχείων που μεταφέρονται. Σκοπός της ανταλλαγής των

κλινικών στοιχείων είναι: η παροχή κλινικής φροντίδας, η υποστήριξη κλινικής και διοικητικής έρευνας, η υποστήριξη των κλινικών δοκιμών και η μεταφορά στοιχείων σε τρίτους φορείς. Για να επιτευχθούν αυτοί οι στόχοι, η Τεχνική Επιτροπή Λεξιλογίου συνεργάζεται και με άλλες ομάδες που δραστηριοποιούνται στα κωδικοποιημένα λεξιλόγια. Μερικές από αυτές τις ομάδες είναι οργανισμοί ανάπτυξης προτύπων, κυβερνητικές αντιπροσωπείες και ρυθμιστικοί οργανισμοί, κλινικές ομάδες ειδικότητας, προμηθευτές λεξιλογίου και προμηθευτές εργαλείων λεξιλογίου.

#### **5.7.4. XML**

*Η* ομάδα ανάπτυξης του HL7 άρχισε να χρησιμοποιεί την τεχνολογία XML από τον Σεπτέμβριο του 1996 όταν σχηματίστηκε η ομάδα ανάπτυξης της SGML/XML. Από τότε, η ομάδα SGML/XML χωρίστηκε σε δύο χωριστές ομάδες: α) την XML Special Interest Group: η οποία υποστηρίζει τον σκοπό ανάπτυξης του HL7 μέσω συστάσεων για τη χρήση των XML προτύπων για όλες τις HL7 πλατφόρμες, β) την Structured Documents Technical Committee: η οποία δραστηριοποιείται στην ανάπτυξη δομημένων προτύπων εγγράφων για την υγειονομική περίθαλψη.

*Οι* δύο αυτές ομάδες έχουν εκπονήσει αρκετά μεγάλη έρευνα η οποία έχει επικυρωθεί από τα μέλη του HL7. Το 1999, ο οργανισμός HL7 επικύρωσε μια σύσταση για τη χρήση της XML ως μέθοδο εναλλακτικής σύνταξης των HL7 μηνυμάτων της έκδοσης V2.3.1. Αυτή η σύσταση ήταν πληροφοριακής φύσης και έτσι δεν υποβλήθηκε στον οργανισμό ANSI προς έγκριση. Εντούτοις, η κωδικοποίηση XML για την έκδοση 2.4 ψηφίστηκε και υποβλήθηκε για έγκριση στον ANSI στις αρχές του 2001.

*Το* Σεπτέμβριο του 2000, τα μέλη του HL7 επικύρωσαν την έκδοση 1 του Clinical Document Architecture, το οποίο καθορίζει μια XML

αρχιτεκτονική για την ανταλλαγή των κλινικών εγγράφων. Η κωδικοποίηση είναι βασισμένη στα XML DTDs και η σημασιολογία της καθορίζεται στο RIM του HL7 καθώς και στα καταχωρημένα κωδικοποιημένα λεξιλόγια. Τον Ιανουάριο του 2005, τα μέλη του HL7 επικύρωσαν την έκδοση 2 του CDA, έτσι η αρχική έκδοση 3 θα χρησιμοποιήσει μόνο την κωδικοποίηση XML. Το HL7 συμμετέχει ενεργά και υποστηρίζει τον W3C, ενός οργανισμού που είναι αρμόδιος για την ανάπτυξη της XML.

## **5.8. ΟΜΑΔΕΣ ΕΡΓΑΣΙΑΣ ΤΟΥ HL7**

### **5.8.1.SIG-HIM: HEALTHCARE INFORMATION MANAGEMENT**

Στόχος της ομάδας αυτής είναι να προτείνει και να τυποποιήσει διαδικασίες όπως: διαδικασίες επίσκεψης σε εξωτερικό ιατρείο, διαδικασίες εισαγωγής και εξόδου από νοσοκομείο, παραγγελία και λήψη εργαστηριακών εξετάσεων, κλπ. Επιπλέον ασχολείται και με την τυποποίηση πληροφοριών όπως δεδομένα του ασθενούς, κατάλογοι υπηρεσιών κλπ.

### **5.8.2. SIG-TIG:TECHNICAL IMPLEMENTATION GUIDELINES**

Αντικείμενο της ομάδας αυτής είναι η τεχνική προσαρμογή των προτύπων του HL7 (εκδόσεις 2.x και 3) στις απαιτήσεις του ελληνικού χώρου καθώς και η ενεργή συμμετοχή στις διαδικασίες διαμόρφωσης του προτύπου σε διεθνές επίπεδο. Στόχος της ομάδας είναι η ανάλυση των προτύπων του HL7 και η προσαρμογή τους στις ανάγκες του ελληνικού συστήματος υγείας. Αυτό επιτυγχάνεται με την υποβολή προτάσεων προς τον οργανισμό HL7 ώστε οι τρέχουσες εκδόσεις των προτύπων να συμπεριλάβουν τις απαραίτητες τροποποιήσεις.

## **5.9. ΠΕΡΙΟΧΕΣ ΕΦΑΡΜΟΓΗΣ ΤΟΥ HL7**

*Παρακάτω φαίνονται οι τομείς εφαρμογής του προτύπου.*

- 1)ADT (εισαγωγή-εξιτήριο-μεταφορά ασθενούς)
- 2)Παραγγελίες ιατρικών πράξεων (order entry):
  - α)Απλές παραγγελίες
  - β)Παραγγελίες φαρμάκων
  - γ)Παραγγελίες Υγ. Υλικού
  - δ)Διατροφή
  - ε)Εργαστηριακές και απεικονιστικές εξετάσεις
- 3)Τιμολόγηση Ασθενών
- 4)Βασικά Αρχεία(MasterFiles-Κωδικοποιήσεις)
- 5)Διαχείριση Ραντεβού
- 6)Αυτοματισμοί εργαστηρίων
- 7)Παραπομπές ασθενών
- 8)Αποτελέσματα Εξετάσεων :
  - Εργαστηριακά και απεικονιστικά
  - Ιατρικά πρωτόκολλα
  - Ιατρικά πορίσματα (CDA)
  - Καρδιογραφήματα κλπ
- 9)Στοιχεία ιατρικού φακέλου (CDA)
- 10)Φροντίδα ασθενών
- 11)Διαχείριση προσωπικού

## **5.10. ΒΑΣΙΚΑ ΣΤΟΙΧΕΙΑ ΤΗΣ ΕΚΔΟΣΗΣ 3**

### **5.10.1. CDA**

Το CDA, γνωστό μέχρι σήμερα ως Patient Record Architecture (PRA), είναι ένα πρότυπο ανταλλαγής κλινικών εγγράφων και φέρνει τη βιομηχανία υγειονομικής περίθαλψης πιο κοντά στην



πραγματοποίηση ενός ηλεκτρονικού ιατρικού αρχείου. Με τη χρήση της τεχνολογίας XML, του RIM και των κωδικοποιημένων λεξιλογίων, το CDA κάνει τα έγγραφα αναγνώσιμα τόσο από τα εμπλεκόμενα συστήματα-έτσι αναλύονται εύκολα για να υποβληθούν σε ηλεκτρονική επεξεργασία-όσο και από τους χρήστες-έτσι μπορούν να ανακτηθούν εύκολα και να χρησιμοποιηθούν από όσους τα χρειάζονται. Τα έγγραφα CDA μπορούν να εκτελεστούν με τη χρήση της XML σε συνδυασμό με Web browsers ή με ασύρματες εφαρμογές όπως τα cell phones.

#### **5.10.2. MDF**

Ο όρος MDF αναφέρεται στην κωδικοποίηση μηνυμάτων ή αρχείων τύπου CDA ώστε να επιτευχθεί επικοινωνία ανάμεσα σε ιατρικά συστήματα. Με λίγα λόγια, προσδιορίζει επακριβώς τη δομή των ανταλλασσόμενων μηνυμάτων μεταξύ των εμπλεκόμενων συστημάτων.

#### **5.10.3. RIM**

Το RIM δίνει μια εικόνα των κλινικών στοιχείων (*Domains*) και προσδιορίζει τον κύκλο ζωής των γεγονότων που ένα μήνυμα ή ομάδες μηνυμάτων μεταφέρουν. Είναι ένα κοινό πρότυπο μεταξύ όλων των domains και κατ' επέκταση το πρότυπο σύμφωνα με το οποίο όλα τα domains δημιουργούν τα μηνύματά τους. Επειδή αντιπροσωπεύει τις συνδέσεις που υπάρχουν ανάμεσα στις πληροφορίες που μεταφέρονται στα πεδία του HL7 μηνύματος μπορεί να θεωρηθεί σαν η αρχιτεκτονική ενός Clinical Document.

#### **5.10.4. CPRS**

Το CPRS παρέχει μία web διεπαφή (δηλαδή παρέχει ένα interface παρόμοιο με ενός web browser) στο UPR. Σε αυτή τη διεπαφή μπορούν να έχουν πρόσβαση όλοι οι εμπλεκόμενοι του χώρου της Υγείας-Ιατρικής, ώστε να τροποποιούν ή να εισάγουν νέα δεδομένα στα υπάρχοντα UPR. Χρησιμοποιείται κλινικό λεξιλόγιο προκειμένου να ανακτηθούν και να αντιστοιχηθούν τα ζητούμενα στοιχεία.

#### **5.10.5. UPR**

Πρόκειται για αρχείο το οποίο περιέχει συγκεντρωμένο όλο το ιατρικό ιστορικό ενός ανθρώπου. Με τη χρήση του CPRS που αναφέραμε, το αρχείο αυτό μπορεί να προσπελαστεί από έναν γιατρό που μπορεί να βρίσκεται σε οποιαδήποτε περιοχή του κόσμου. Ουσιαστικά πρόκειται για έναν ηλεκτρονικό ιατρικό φάκελο.

### **5.11. ΓΕΝΙΚΗ ΕΠΙΣΚΟΠΗΣΗ ΤΟΥ ΠΡΟΤΥΠΟΥ**

#### **5.11.1. ΕΙΣΑΓΩΓΗ**

Στη συνέχεια θα παρουσιάσουμε μια περιγραφή του τρόπου σκέψης που ακολουθήθηκε στην ανάπτυξη του πρωτοκόλλου HL7, στην επεξεργασία των μεταβλητών και του τρόπου δόμησης του, ώστε να μπορεί να αντεπεξέλθει σε μελλοντικές αλλαγές.

#### **5.11.2. ΚΑΝΟΝΕΣ ΚΩΔΙΚΟΠΟΙΗΣΗΣ ΤΟΥ HL7**

Τα μηνύματα του HL7 αποτελούνται από πεδία δεδομένων μεταβλητού μεγέθους που διαχωρίζονται από το χαρακτήρα διαχωρισμού πεδίου (field separator character). Οι κανόνες περιγράφουν τον τρόπο κωδικοποίησης διαφόρων τύπων δεδομένων σε ένα πεδίο καθώς και πότε το πεδίο μπορεί να επαναληφθεί. Τα πεδία δεδομένων με τη σειρά τους συνδυάζονται σε λογικές ομάδες

που ονομάζονται κομμάτια (segments). Τα κομμάτια είναι διαχωρισμένα με χαρακτήρες διαχωρισμού κομματιών (segment separator characters). Κάθε κομμάτι ξεκινά με μια τιμή τριών χαρακτήρων που το χαρακτηρίζει μέσα στο μήνυμα. Τα κομμάτια δύναται να είναι υποχρεωτικά ή προαιρετικά και μπορούν να επαναληφθούν.

Τα δεδομένα χρησιμοποιούν το σύνολο χαρακτήρων ASCII (περιλαμβάνει δεκαεξαδικές τιμές μεταξύ 20 και 7E) εκτός και αν δηλώνεται κάτι άλλο στο MSH κομμάτι του μηνύματος. Ο χαρακτήρας διαχωρισμού πεδίου χρειάζεται να επιλεγεί από το σύνολο ASCII. Όλοι οι υπόλοιποι ειδικοί χαρακτήρες μπορούν να επιλεγούν από άλλα σύνολα, εκτός του χαρακτήρα διαχωρισμού κομματιού που είναι ο ASCII Carriage Return χαρακτήρας.

Στην Ευρωπαϊκή Κοινότητα του HL7, προέκυψε η ανάγκη για εκτυπώσιμους χαρακτήρες οι οποίοι δεν συμπεριλαμβάνονται στο πιο πάνω σύνολο. Οι χαρακτήρες αυτοί παίρνουν τιμές μεταξύ 128 και 256. Αυτοί οι χαρακτήρες εμπεριέχονται στο ISO 8859, πρόκειται για σύνολο 256 χαρακτήρων που περιλαμβάνει όλους τους χαρακτήρες που χρειάζεται η Ευρωπαϊκή Κοινότητα.

### **5.11.3. ΚΩΔΙΚΕΣ ΠΟΛΛΩΝ ΧΑΡΑΚΤΗΡΩΝ**

Όταν υπάρχει ανάγκη για χρήση κωδικών μεγάλου μήκους, τότε η επεξεργασία τους γίνεται με έναν από τους παρακάτω τρόπους:

- UNICODE: Όταν τα συνδιαλεγόμενα μέρη χρησιμοποιούν UNICODE και όλοι οι χαρακτήρες χρησιμοποιούν τον ίδιο αριθμό bytes τότε όλοι οι οριοθέτες (delimiters) θα είναι χαρακτήρες του ίδιου μήκους. Το πρότυπο λοιπόν θα τους αναγνωρίζει σαν να

ήταν χαρακτήρες ενός byte με τη διαφορά ότι το μήκος σε bytes των χαρακτήρων μπορεί να είναι μεγαλύτερο.

- JIS X 0202 – ISO 2022: Εδώ υπάρχει η δυνατότητα παροχής μίας ακολουθίας διαφυγής (escape sequence) ούτως ώστε να γίνεται η μεταφορά από το ένα σύνολο χαρακτήρων στο άλλο.
- Αν τέλος, μια χώρα που χρησιμοποιεί κώδικες μεγάλου μήκους δεν εμπίπτει σε κανέναν από τους παραπάνω τρόπους επεξεργασίας των κωδικών μεγάλου μήκους τότε μπορεί αυθαίρετα να ορίσει κάποιους κανόνες για τον τρόπο επικοινωνίας των δύο μερών.

#### **5.11.4. ΑΝΤΙΜΕΤΩΠΙΣΗ ΔΙΑΦΟΡΟΠΟΙΗΣΕΩΝ**

Το HL7 μοντελοποιεί τη μεταφορά δεδομένων και όχι τα εκάστοτε συστήματα που το χρησιμοποιούν. Κάτι που συνεπάγεται μεγάλη ποικιλία στον τρόπο εφαρμογής του προτύπου ανά τον κόσμο. Παρά όμως τους ποικίλους τρόπους εφαρμογής του, υπάρχουν κάποιοι κανόνες που παραμένουν αμετάβλητοι. Παραδείγματος χάριν, τα πεδία δεδομένων που είναι απαραίτητα σε ένα θεωρητικό μήνυμα είναι εκείνα που υποστηρίζουν τη σχεσιακή λογική μεταξύ μηνυμάτων ή τη λειτουργία αυτών. Τα υπόλοιπα πεδία είναι προαιρετικά και μπορεί να μην αναφερθούν ανάλογα με την επιλογή του τρόπου εφαρμογής τους. Τέλος, υπάρχουν κανόνες και περιορισμοί που ορίζουν την πιθανή δημιουργία νέων μηνυμάτων σε κάθε χώρα, ώστε να εξασφαλιστεί και η λειτουργία μελλοντικών εκδόσεων του προτύπου εκεί.

#### **5.11.5. ΝΕΕΣ ΕΚΔΟΣΕΙΣ ΤΟΥ ΠΡΟΤΥΠΟΥ**

Το πρότυπο συνεχώς ανανεώνεται με την κυκλοφορία νέων εκδόσεων αυτού. Γι αυτό το λόγο όλα τα μηνύματα που στέλνονται μέσω του

προτύπου, περιλαμβάνουν και έναν αριθμό της έκδοσης του προτύπου (version ID).

Όταν σε ένα μήνυμα προστεθούν κάποια πεδία μιας νεότερης έκδοσης από την τρέχουσα τότε, το περιεχόμενο αυτών των πεδίων αγνοείται από την καλούσα εφαρμογή μέχρις ότου αναβαθμιστεί και η τρέχουσα έκδοση.

#### **5.11.6. ΜΕΤΑΦΟΡΑ ΑΡΧΕΙΩΝ**

Το HL7 μπορεί πολύ εύκολα να χρησιμοποιηθεί και για μεταφορά αρχείων λόγω του ότι λειτουργεί σε εφαρμογές τύπου client-server. Τα επιθυμητά μηνύματα του αρχείου κωδικοποιούνται, ομαδοποιούνται και μεταφέρονται χρησιμοποιώντας εξωτερικά μέσα όπως τα FTAM, FTP.

#### **5.11.7. ΣΥΝΕΡΓΑΣΙΑ ΜΕ ΑΛΛΑ ΠΡΩΤΟΚΟΛΛΑ**

Το πρωτόκολλο HL7 συνεργάζεται και με άλλα πρωτόκολλα, τα οποία είναι:

1. ACR/NEMA DICOM.
2. IEEE P1157 ("MEDIX").
3. ASC X12 Standards for Electronic Document Interchange.
4. ASTM 1238.94 Laboratory Data Reporting.

#### **5.12. ΓΕΓΟΝΟΤΑ ΩΘΗΣΗΣ - ΣΚΑΝΔΑΛΗΣ (TRIGGER EVENTS)**

Ο τρόπος γραφής του πρωτοκόλλου ακολουθεί την υπόθεση ότι ένα γεγονός από τον πραγματικό κόσμο των υπηρεσιών υγείας, δημιουργεί την ανάγκη να μεταφερθούν δεδομένα μεταξύ συστημάτων. Αυτό το γεγονός ονομάζεται «γεγονός ώθησης-σκανδάλης».

Όταν η μετάδοση μιας πληροφορίας ξεκινήσει με την έναρξη του γεγονότος σκανδάλης, η συναλλαγή ονομάζεται «αυτόκλητη ενημέρωση» (unsolicited update).

Το πρότυπο HL7 επιτρέπει την χρήση γεγονότων σκανδάλης που ενδέχεται να έχουν διαφορετικά επίπεδα στα είδη των δεδομένων που αποστέλλουν. Μπορεί να συσχετίζονται μπορεί και όχι.

### **5.13. ACKNOWLEDGMENTS**

Ο τρόπος επιβεβαίωσης (acknowledgment) μιας αυτόκλητης ενημέρωσης που στέλνεται από ένα σύστημα σε ένα άλλο, μας δείχνει ότι επιβεβαιώθηκε η ενημέρωση στο επίπεδο εφαρμογής. Ένα τέτοιο παράδειγμα είναι η παραγγελία κάποιας εργαστηριακής εξέτασης για κάποιον ασθενή. Το σύστημα που θα λάβει την παραγγελία θα απαντήσει ότι την έχει δεχθεί μαζί με όποιες πληροφορίες έρχονται για τον ασθενή. Μπορεί στην απάντηση να στείλει και κάποιον αριθμό που δείχνει την σειρά παραλαβής του.

Το πρότυπο δεν ορίζει τις ενέργειες που πρέπει να γίνουν όταν η εκάστοτε εφαρμογή λάβει ένα μήνυμα, καθώς επίσης δεν μπορεί να εξάγει συμπεράσματα για την ιδιοκτησία των δεδομένων του μηνύματος.

### **5.14. ACKNOWLEDGMENTS: ENHANCED MODE**

Όταν λαμβάνεται ένα μήνυμα η εφαρμογή είναι σε θέση όχι μόνο να επιβεβαιώσει την ορθή λήψη τους αλλά και να στείλει μια απάντηση η οποία δείχνει ότι το μήνυμα επεξεργάστηκε. Μόλις το μήνυμα ληφθεί, η εφαρμογή στέλνει μήνυμα ότι το έλαβε και στη συνέχεια

όταν το επεξεργαστεί στέλνει μία επιβεβαίωση ότι το επεξεργάστηκε και την κατάσταση στην οποία βρίσκεται το μήνυμα.

#### **5.15. ΠΕΡΙΒΑΛΛΟΝΤΑ ΕΠΙΚΟΙΝΩΝΙΑΣ**

Τα περιβάλλοντα επικοινωνίας στα οποία μπορεί να εφαρμοστεί το HL7 είναι αρκετά. Μπορεί να είναι ένα ad-hoc περιβάλλον, που δεν προσφέρει εγγυήσεις μεταφοράς, ένα αξιόπιστο περιβάλλον όπως το TCP/IP, το DECNET και το SNA, τα οποία δεν έχουν ιδιαίτερα υψηλές απαιτήσεις ή ακόμα ένα υψηλότερων απαιτήσεων περιβάλλον όπως το IBM SNA LU6.2 και το Sun Microsystems NFS.

Το πρότυπο όμως για να λειτουργήσει σωστά προϋποθέτει τα παρακάτω από το περιβάλλον επικοινωνίας:

1. Μετάδοση χωρίς λάθη. Αυτό σημαίνει ότι γίνεται έλεγχος λαθών σε κατώτερο επίπεδο.
2. Μετατροπή χαρακτήρων. Αν δύο μηχανές που ανταλλάσσουν δεδομένα χρησιμοποιούν διαφορετική κωδικοποίηση δεδομένων, το περιβάλλον επικοινωνίας πρέπει να κάνει σωστή μετατροπή των χαρακτήρων.
3. Μέγεθος μηνύματος. Το πρότυπο δεν βάζει όρια στο μέγεθος του μηνύματος που θα αποσταλεί. Προϋποθέτει ότι το περιβάλλον επικοινωνίας μπορεί να μεταδώσει οποιουδήποτε μεγέθους μηνύματα ή ότι θα χρησιμοποιήσει το πρωτόκολλο συνέχειας μηνύματος του HL7.

#### **5.16. ΚΑΝΟΝΕΣ ΔΗΜΙΟΥΡΓΙΑΣ ΤΩΝ HL7 ΜΗΝΥΜΑΤΩΝ**

Οι κανόνες σύνταξης των HL7 μηνυμάτων περιγράφονται αναλυτικά στο πρότυπο HL7 και δημιουργούν μηνύματα αυθαίρετου μήκους. Παρακάτω παραθέτουμε τον ψευδοκώδικα κατασκευής των μηνυμάτων.

## Message Construction Pseudocode

```
procedure transmit_message ( data ) {
  identify_message_needed;
  validate( data );
  order_segments( data, segment_list );
  foreach segment in ( segment_list ) {
    print segment.name; /* e.g., MSH */
    /* gather all data for fields */
    foreach field in ( fields_of( segment ) ) {
      print field separator; /* e.g., | */
      /* gather occurrences (may be multiple only for fields that are
allowed to repeat */
      foreach occurrence in ( occurrences_of( field ) ) {
        transmit_occurrence( occurrence );
        if not last ( populated occurrence ) print repetition_separator;
/* e.g., ~ */
      }
      break if last ( populated field );
    }
    print segment_terminator;          /* always<cr>! */
  }

  return;
}

procedure transmit_occurrence ( occurrence ) {
```



```

/* gather populated components */
foreach component in ( components_of( occurrence ) ) {
    get_subcomponent_data( component );

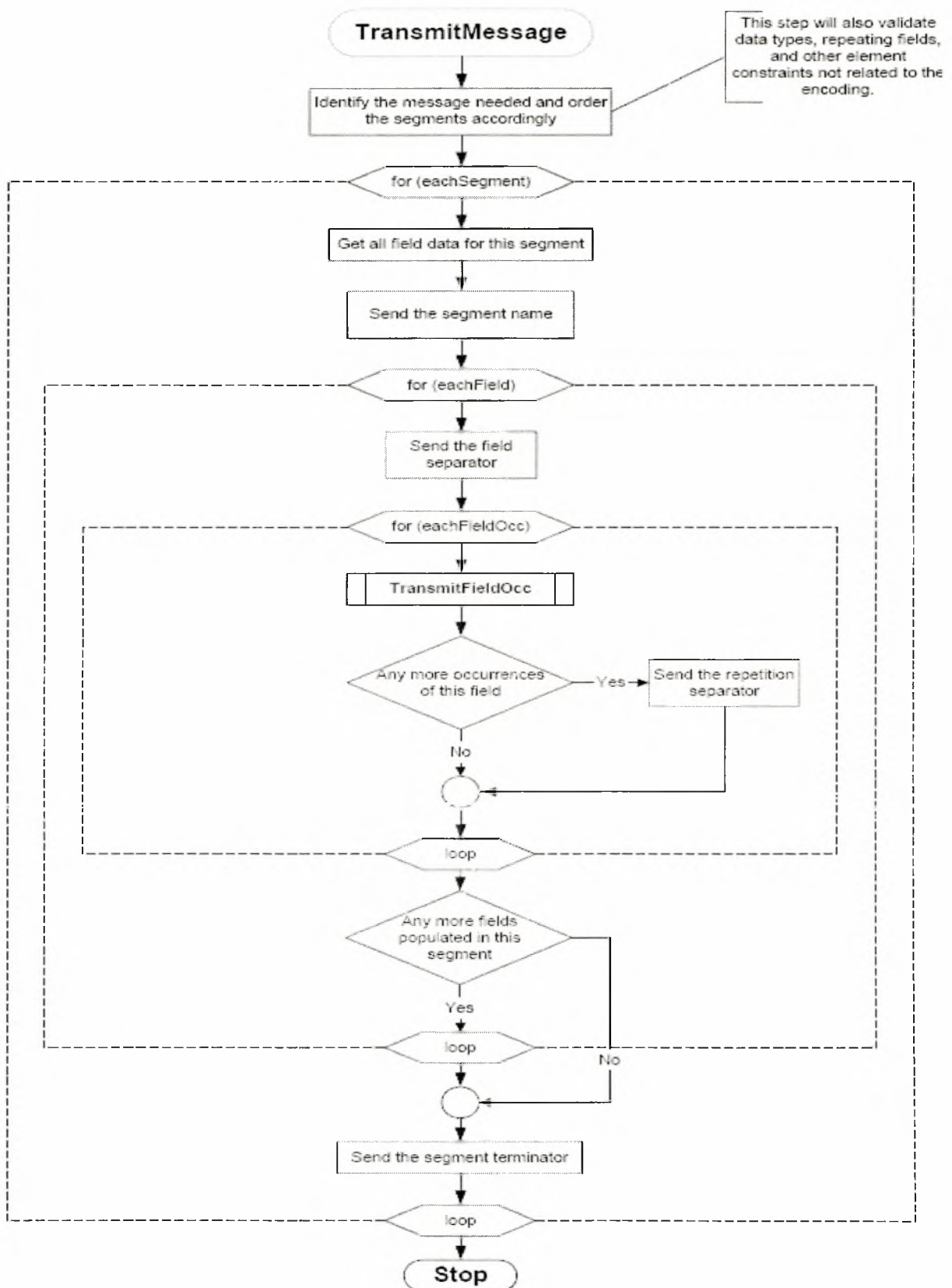
    /* gather all data for subcomponents */
    foreach subcomponent in ( subcomponents_of( component ) )
    {
        /* escape the field separator */
        substitute( field_separator, \F\ );
        /* escape the encoding characters */
        substitute( component_separator, \S\ );
        substitute( repetition_separator, \R\ );
        substitute( escape_character, \E\ );
        substitute( subcomponent_separator, \T\ );
        print subcomponent;
        if not last ( populated subcomponent ) print
subcomponent_separator; /* e.g., & */
    }

    if not last ( populated component ) print component_separator;
/* e.g., ^ */
}
return;
}

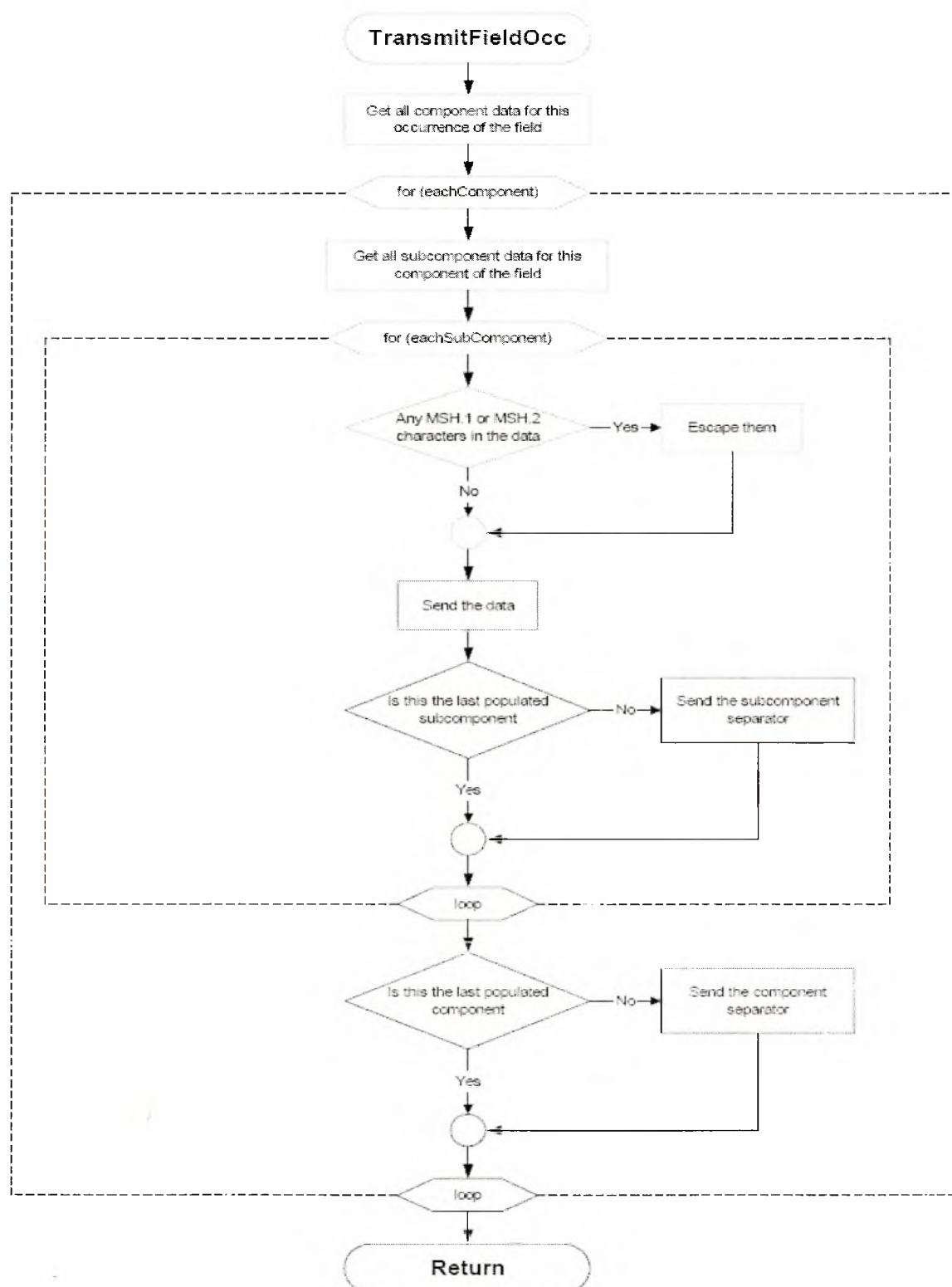
```

Στη συνέχεια παραθέτουμε δύο διαγράμματα ροής που αναπαριστούν τους κανόνες κατασκευής μηνύματος. Το πρώτο μας δείχνει τους κανόνες αποστολής ενός μηνύματος και το δεύτερο τις εμφανίσεις των πεδίων αυτού.

## Διάγραμμα ροής κατασκευής μηνύματος (1)



## Διάγραμμα ροής αποστολής μηνύματος (2)



## **5.17. ΠΕΡΙΓΡΑΦΗ ΤΩΝ ΕΦΑΡΜΟΓΩΝ ΤΟΥ HL7**

### **5.17.1. ΑΝΑΠΑΡΑΣΤΑΣΗ ΜΗΝΥΜΑΤΩΝ ΣΤΟ HL7**

Κάθε μήνυμα στο HL7 ορίζεται με ένα ειδικό συμβολισμό που δείχνει όλα τα αναγνωριστικά των segment με την σειρά που πρέπει να εμφανιστούν στο μήνυμα. Τα άγκιστρα {...}, ορίζουν ότι η ομάδα των segment που βρίσκεται μέσα σε αυτά μπορεί να επαναληφθεί μία ή περισσότερες φορές. Οι αγκύλες [...] δείχνουν ότι η ομάδα που βρίσκεται μέσα σε αυτές είναι προαιρετική (optional). Αν κάποια segment είναι προαιρετικά και μπορούν να επαναληφθούν τότε, θα είναι σε αγκύλες και άγκιστρα ανεξάρτητα με την σειρά που θα τοποθετηθούν αυτές. Δηλαδή το {...} είναι το ίδιο με το [{...}]

Η επιλογή ενός segment από μια ομάδα segments γίνεται με τη χρησιμοποίηση γωνιών <...> και με κάθετες γραμμές'|' για να ξεχωρίζουν τα διαφορετικά segments.

### **5.17.2. PATIENT ADMINISTRATION – ΔΙΑΧΕΙΡΙΣΗ ΑΣΘΕΝΩΝ**

Σε αυτές τις παραγράφους θα παρουσιάσουμε κάποια γεγονότα-σκανδάλης (trigger events) που ενδέχεται να συμβούν σε μια μονάδα παροχής ιατρικών υπηρεσιών. Τα γεγονότα αυτά περιγράφουν την επίβλεψη των ασθενών. Είναι κυρίως δημογραφικά στοιχεία του ασθενούς και στοιχεία σχετικά με την επίσκεψη αυτού σε μία μονάδα. Τα στοιχεία αυτά είναι απαραίτητα από άλλα υποσυστήματα μιας υγειονομικής υπηρεσίας. Οι συναλλαγές που αφορούν σε αυτά τα δεδομένα συμβαίνουν στο έβδομο επίπεδο, δηλαδή στα αφηρημένα μηνύματα. Πρέπει να ξεκαθαρίσουμε ότι τα γεγονότα ώθησης-σκανδάλης ποικίλουν στα διάφορα νοσοκομειακά συστήματα.

Η ερμηνεία των μηνυμάτων αυτών των γεγονότων βασίζεται στην κατηγορία του ασθενούς (patient class). Η πληροφορία που περιέχεται σε οποιαδήποτε από αυτά τα μηνύματα μπορεί να είναι περισσότερη από την ελάχιστη απαιτούμενη. Οποιοδήποτε από τα πεδία που είναι μέσα στα κομμάτια (segments) του μηνύματος μπορεί να χρησιμοποιηθεί. Μπορεί όμως να καθοριστεί μέσα από το σύστημα ο αριθμός των πεδίων που πρέπει να χρησιμοποιηθούν. Για να εξαλειφθεί αυτή η ασάφεια, το πρότυπο προτείνει (αλλά δεν απαιτεί) το γεγονός A08 (ενημέρωση πληροφοριών ασθενούς-update patient information). Αυτό χρησιμοποιείται για να ενημερώνει τα πεδία που δε συνδέονται με άλλα γεγονότα ώθησης.

Όλα τα μηνύματα κατασκευάστηκαν με τους κανόνες που περιγράφονται στο HL7. Ορισμένα από αυτά τα μηνύματα είναι:

- **ADT/ACK – Admit/Visit Notification ( Event A01):** Εισαγωγή /Επίσκεψη Ασθενούς. Η πληροφορία που μεταφέρει αυτό το μήνυμα χρησιμοποιείται για «εσωτερικούς ασθενείς» (admitted patients) μόνο. Σηματοδοτεί την εισαγωγή ασθενούς σε μια ιατρική μονάδα και την τοποθέτηση του σε κάποιο κρεβάτι. Η πληροφορία εισάγεται στο κυρίως Σύστημα Διαχείρισης Ασθενών και εκπέμπεται σε όλα τα νοσοκομειακά και βοηθητικά συστήματα.
- **ADD PERSON OR PATIENT INFORMATION:** Πρόσθεση προσώπου ή πληροφορίες ασθενούς.
- **CANCEL PATIENT DEPARTING – TRACKING:** Ακύρωση αναχώρησης ασθενούς.
- **MOVE PATIENT INFORMATION/PATIENT IDENTIFIER LIST:** Μεταφορά πληροφοριών ασθενούς/Λίστα ταυτοποίησης ασθενούς.

Στη συνέχεια παραθέτουμε έναν πίνακα που δείχνει τον ορισμό των μηνυμάτων αυτών όπως ορίζονται στο πρότυπο HL7.

Παράδειγμα αυτού είναι το κομμάτι EVN – Event Type Segment που περιγράφεται από το πρότυπο:

HL7 Attribute Table - EVN – Event Type							
SEQ	LEN	DT	OPT	RP#	TBL#	ITEM#	ELEMENT NAME
1	3	ID	B		0003	00099	Event Type Code
2	26	TS	R			00100	Recorded Date/Time
3	26	TS	O			00101	Date/Time Planned Event
4	3	IS	O		0062	00102	Event Reason Code
5	250	XCN	O	Y	0188	00103	Operator ID
6	26	TS	O			01278	Event Occurred
7	241	HD	O			01534	Event Facility

Βλέπουμε ότι κάθε μέρος του κομματιού είναι καθορισμένο και δίνεται η σειρά του, το μήκος του, το είδος των δεδομένων (Data Type- DT), πληροφορία για το αν μπορεί το κάθε σημείο να επαναληφθεί (RP#), η ύπαρξη πίνακα που δίνει κάποιες παραμέτρους για το σημείο αυτό (TBL#), ένας αύξων αριθμός που το καθορίζει και μια σύντομη περιγραφή του.

Βλέπουμε ότι ο πίνακας που περιγράφει το segment EVN αναφέρεται στα datatype IS και XCN, τα οποία με τη σειρά τους αναφέρονται σε δύο άλλους πίνακες με προκαθορισμένες τιμές.

### **5.17.3. ORDER ENTRY – ΕΙΣΑΓΩΓΗ ΠΑΡΑΓΓΕΛΙΩΝ**

Σε αυτή την εφαρμογή γίνεται προτυποποίηση των παραγγελιών. Με τον όρο παραγγελίες εννοούμε αιτήσεις για υλικά ή υπηρεσίες, πληροφορίες για παραγγελίες μεταξύ τμημάτων της νοσοκομειακής μονάδας (υλικά, υπηρεσίες που συνήθως αφορούν κάποιον ασθενή

), αιτήσεις για φαρμακευτικά προϊόντα, κλινικές παρατηρήσεις, εξετάσεις από εργαστήρια, φαγητό και πολλά άλλα.

Αυτή η εφαρμογή χωρίζεται σε έξι επιμέρους τμήματα: Γενικό, Διαιτητικό, Παροχής, Φαρμακείο, Εμβολιασμού και Μεταγγίσεως. Το κάθε τμήμα περιέχει τα γεγονότα ώθησης-σκανδάλης για την αποστολή μηνύματος, τον ορισμό του μηνύματος, τα κομμάτια και παραδείγματα παραγγελιών. Η γενική περιγραφή μηνυμάτων είναι η ίδια που χρησιμοποιείται και στα άλλα κεφάλαια. Μερικά παραδείγματα μηνυμάτων που αντιστοιχούν σε αυτή την εφαρμογή είναι:

- **ORM-GENERAL ORDER MESSAGE (EVENT O01):** Γενικό μήνυμα παραγγελίας.
- **ORR-GENERAL ORDER RESPONSE MESSAGE RESPONSE TO ANY ORM (EVENT Q02):** Γενική απάντηση παραγγελίας σε οποιοδήποτε μήνυμα παραγγελίας.
- **OSQ/OSR-QUERY RESPONSE FOR ORDER STATUS (EVENT Q06):** Απάντηση αίτησης για αναφορά της κατάστασης της παραγγελίας.
- **OMG-GENERAL CLINICAL ORDER MESSAGE (EVENT O19):** Γενικό μήνυμα κλινικής παραγγελίας.
- **ORG – GENERAL CLINICAL ORDER ACKNOWLEDGEMENT MESSAGE (EVENT O20):** Γενική αποδοχή κλινικής παραγγελίας.
- **OML-LABORATORY ORDER MESSAGE (EVENT O21):** Μήνυμα παραγγελίας εργαστηρίου.
- **OMP-PHARMACY/TREATMENT ORDER MESSAGE (EVENT O09):** Παραγγελία φαρμακείου/προϊόντων περίθαλψης.
- **ORP – PHARMACY/TREATMENT ORDER ACKNOWLEDGMENT (EVENT O10):** Αποδοχή παραγγελίας φαρμακείου/ προϊόντων περίθαλψης.

- **RDE - PHARMACY/TREATMENT ENCODED ORDER MESSAGE (EVENT O11):** Κωδικοποιημένη παραγγελία φαρμακείου/ προϊόντων περίθαλψης.

#### **5.17.4. FINANCIAL MANAGEMENT –ΟΙΚΟΝΟΜΙΚΗ ΔΙΑΧΕΙΡΙΣΗ**

*Η εφαρμογή αυτή, όπως φανερώνει και το όνομα της αφορά στις οικονομικές συναλλαγές του HL7. Οι συναλλαγές αυτές γίνονται συνολικά ή online μία προς μία. Το πρότυπο χρησιμοποιεί κανονισμούς που διέπουν τις συναλλαγές και που κυρίως χρησιμοποιούνται στις Ηνωμένες Πολιτείες Αμερικής. Υπάρχουν όμως και εφαρμογές των κανονισμών αυτών, όπως είναι ο αριθμός κοινωνικών ασφαλίσεων, που υπάρχουν και σε άλλες χώρες. Υπάρχει μεγάλη ποικιλία συναλλαγών, μερικές εκ των οποίων παρουσιάζονται στη συνέχεια.*

- **BAR/ACK - ADD PATIENT ACCOUNT (EVENT P01):** Άνοιγμα Λογαριασμού Ασθενούς.
- **BAR/ACK - PURGE PATIENT ACCOUNTS (EVENT P02):** Διαγραφή Λογαριασμού Ασθενούς.
- **DFT/ACK - POST DETAIL FINANCIAL TRANSACTIONS (EVENT P03):** Λεπτομέρειες Συναλλαγών.
- **QRY/DSR - GENERATE BILLS AND ACCOUNTS RECEIVABLE STATEMENTS (EVENT P04):** Δημιουργία Αποδείξεων.
- **BAR/ACK - UPDATE ACCOUNT (EVENT P05):** Ενημέρωση Λογαριασμών.
- **DFT/ACK - POST DETAIL FINANCIAL TRANSACTIONS - EXPANDED (EVENT P11):** Περισσότερες Λεπτομέρειες Συναλλαγών.



### **5.17.5. QUERY – ΑΙΤΗΣΕΙΣ**

*Εδώ ορίζονται οι κανόνες που εφαρμόζονται στις εκάστοτε αιτήσεις και στις απαντήσεις τους. Τέλος καθορίζεται η μορφή των αυτόκλητων μηνυμάτων (unsolicited display messages).*

*Μερικά τυπικά παραδείγματα των συνηθέστερων αιτήσεων που ενδέχεται να συμβούν είναι:*

- Δεδομένα ενός ασθενούς π.χ. αποστολή των δεδομένων του ασθενούς #123456.
- Δεδομένα πολλών ασθενών π.χ. αποστολή όλων των ασθενών που επισκέφθηκαν τον γιατρό δρ.#123.
- Δεδομένα που δε σχετίζονται με ασθενή π.χ. αποστολή της σύστασης ενός φαρμάκου.

*Είναι προφανές ότι δεν υπάρχει κάποιο όριο στον αριθμό των αιτήσεων που μπορεί να αποσταλούν. Η εφαρμογή αυτή πραγματεύεται διάφορους γενικούς τρόπους με τους οποίους μπορεί να υλοποιηθούν αιτήσεις και απαντήσεις. Τα λειτουργικά κεφάλαια πραγματεύονται συγκεκριμένες αιτήσεις/απαντήσεις.*

*Ενδεικτικά ορισμένα μηνύματα που ανταλλάσσονται είναι:*

- **QBP/RDY – QUERY BY PARAMETER/DISPLAY RESPONSE (EVENTS VARY):** Απάντηση αίτησης για παραμέτρους.
- **QVR - QUERY FOR PREVIOUS EVENTS (EVENT Q17):** Αίτηση για προηγούμενα γεγονότα.
- **QCN/ACK - CANCEL QUERY/ACKNOWLEDGE MESSAGE (EVENT J01):** Ακύρωση μηνύματος αίτησης/αποδοχής.

#### **5.17.6. OBSERVATION REPORTING-ΑΝΑΦΟΡΑ ΠΑΡΑΤΗΡΗΣΕΩΝ**

*Η εφαρμογή αυτή αφορά στους κανόνες συναλλαγής που απαιτούνται για την αποστολή κλινικών δεδομένων από έναν υπολογιστή σε έναν άλλον. Η πιο συνηθισμένη χρήση αυτών των κανόνων είναι η αποστολή παρατηρήσεων και αποτελεσμάτων διαγνώσεων προς το σύστημα που τις ζητά (π.χ. το σύστημα στο γραφείο του γιατρού). Τα αποτελέσματα και οι παρατηρήσεις μπορούν επίσης να σταλούν από το αρχικό σύστημα και σε άλλα συστήματα που πιθανώς τα χρειάζονται, εκτός αυτών που τα ζήτησαν.*

*Ακόμα, η εφαρμογή αυτή σχετίζεται και με την παροχή μεθοδολογίας για τη διασύνδεση παραγγελιών και τη μεταφορά κλινικών παρατηρήσεων, κλινικών εργαστηριακών εξετάσεων, μέτρων της κατάστασης του ασθενούς, ζωτικών μετρήσεων, δοσοληψιών φαρμάκων.*

*Παραθέτουμε ορισμένα ενδεικτικά παραδείγματα:*

- **ORU – UNSOLICITED OBSERVATION MESSAGE (EVENT R01):** Αυτέκλητο μήνυμα παρατηρήσεων.
- **QRY/ORF - QUERY FOR RESULTS OF OBSERVATION (EVENTS R02, R04):** Αίτηση για αποτελέσματα παρατηρήσεων.

#### **5.17.7. MEDICAL RECORDS/INFORMATION MANAGEMENT – ΔΙΑΧΕΙΡΙΣΗ ΑΡΧΕΙΩΝ ΙΑΤΡΙΚΩΝ ΠΛΗΡΟΦΟΡΙΩΝ**

*Η* εφαρμογή αυτή προς το παρόν υποστηρίζει μόνο την διαχείριση κειμένων (document management). Μελλοντικά αναμένεται να υποστηρίξει τις ανάγκες δεδομένων και άλλων εφαρμογών. Σκοπός του ιατρικού αρχείου είναι να παρέχει ένα ακριβές, νόμιμο και ευανάγνωστο κείμενο όλων των υπηρεσιών που έλαβε ο ασθενής. Ακολουθούν παραδείγματα:

- **MDM/ACK - ORIGINAL DOCUMENT NOTIFICATION (EVENT T01):** Ειδοποίηση αυθεντικού αρχείου.
- **MDM/ACK - ORIGINAL DOCUMENT NOTIFICATION AND CONTENT (EVENT T02):** Ειδοποίηση και περιεχόμενο αυθεντικού αρχείου.
- **MDM/ACK - DOCUMENT STATUS CHANGE NOTIFICATION (EVENT T03):** Ειδοποίηση αλλαγής κατάστασης αρχείου.
- **MDM/ACK - DOCUMENT ADDENDUM NOTIFICATION (EVENT T05):** Ειδοποίηση προσάρτησης αρχείου.

#### **5.17.8. SCHEDULING – ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ**

*Οι* εφαρμογές εδώ περιλαμβάνουν αφηρημένα μηνύματα για την αποστολή γεγονότων σχετικά με τον προγραμματισμό επισκέψεων σε κάποιον γιατρό ή για την χρησιμοποίηση πόρων της κλινικής. Υπάρχουν τρεις βασικοί τύποι μηνυμάτων:

1. Απαίτηση Συναλλαγών και Απαντήσεων (request transactions): Συμβαίνουν όταν κάποιος κάνει αίτηση για να ζητήσει μια υπηρεσία και δεν είναι σαφής ο τρόπος εκπλήρωσης της.
2. Αίτηση Συναλλαγών και Απαντήσεων (query transactions): Χρησιμοποιούνται για την ανταλλαγή πληροφορίας μεταξύ συστημάτων προκειμένου να γίνει σωστός προγραμματισμός.

3. Αυτόκλητες Συναλλαγές και Απαντήσεις (unsolicited transactions): Χρησιμοποιούνται για την ανταλλαγή πληροφορίας μεταξύ των συστημάτων προκειμένου να γίνει ο σωστός προγραμματισμός.

Η ανταλλαγή αυτών των πληροφοριών μπορεί να γίνει είτε ενεργητικά είτε παθητικά. Η ενεργητική συλλογή πληροφοριών γίνεται με την έκδοση αιτήσεων συναλλαγών ανάμεσα σε δύο εφαρμογές (η μία καλείται να την εκπληρώσει και η άλλη τη ζητά). Η παθητική συλλογή πληροφοριών γίνεται με την αποδοχή αυτόκλητων συναλλαγών από μια εφαρμογή που τη ζητά. Κάποια μηνύματα που είναι πιθανό να δημιουργηθούν είναι:

- **REQUEST NEW APPOINTMENT BOOKING (EVENT S01):**  
Αίτηση κράτησης νέας ημερομηνίας επίσκεψης.
- **REQUEST APPOINTMENT RESCHEDULING (EVENT S02):** Αίτηση αλλαγής ημερομηνίας επίσκεψης.
- **NOTIFICATION OF APPOINTMENT RESCHEDULING (EVENT S13):** Ειδοποίηση αλλαγής ημερομηνίας επίσκεψης.

#### **5.17.9. MASTER FILES – ΑΡΧΕΙΑ ΔΙΑΧΕΙΡΙΣΤΗ**

Στα συστήματα που βρίσκονται σε περιβάλλοντα ανοιχτής αρχιτεκτονικής κρατείται ένα σύνολο αρχείων που χρησιμοποιούνται ευρέως από ένα ή περισσότερα συστήματα. Αυτά τα αρχεία καλούνται αρχεία διαχείρισης. Μερικά παραδείγματα τέτοιων αρχείων είναι:

1. Αρχείο διαχείρισης προσωπικού και ιατρών.
2. Αρχείο διαχείρισης χρήστη συστήματος (password).
3. Αρχείο διαχείρισης τοποθεσίας (έρευνας και κλινικής).
4. Αρχείο διαχείρισης τερματικών, εκτυπωτών, σταθμών εργασίας κτλ.

5. Κωδικοί εξετάσεων (ακτινολογίας) .
6. Αρχείο διαχείρισης χρεώσεων.
7. Αρχείο διαχείρισης κατάστασης ασθενούς.
8. Αρχείο διαχείρισης τύπου ασθενούς.
9. Αρχείο διαχείρισης υπηρεσιών.

*Προκειμένου να γίνεται σωστή χρήση αυτών των αρχείων πρέπει να υπάρχει συγχρονισμός μεταξύ των εφαρμογών, ειδικότερα όταν αυτές εκτελεσθούν ταυτόχρονα. Το μήνυμα για την ειδοποίηση των Αρχείων Διαχειριστή (Master Files Notification message) παρέχει ένα τρόπο να διατηρείται αυτός ο συγχρονισμός.*

*Παραθέτουμε μερικά παραδείγματα μηνυμάτων:*

- **MASTER FILE NOTIFICATION - TEST/OBSERVATION (EVENT M03):** Ειδοποίηση αρχείου διαχειριστή.
- **MFN/MFK - MASTER FILE NOTIFICATION - TEST/OBSERVATION (NUMERIC) (EVENT M08):** Ειδοποίηση αρχείου διαχειριστή (αριθμητικό).

#### **5.17.10. PATIENT REFERRAL – ΠΑΡΑΠΕΜΠΤΙΚΑ ΑΣΘΕΝΟΥΣ**

*Οι εφαρμογές εδώ, ορίζουν τα σύνολα των μηνυμάτων που θα μεταβιβαστούν ανάμεσα σε διαφορετικές ιατρικές υπηρεσίες. Αυτές οι συναλλαγές συμβαίνουν μεταξύ συστημάτων με διαφορετικές μεθόδους απόκτησης και αποθήκευσης δεδομένων. Η διαθεσιμότητα, πληρότητα και ο βαθμός ενημέρωσης για κάποιον ασθενή παρουσιάζει πολλές διαφοροποιήσεις.*

*Τα γεγονότα στον πραγματικό κόσμο που ωθούν την πραγματοποίηση τέτοιων μηνυμάτων δεν περιορίζονται σε ένα*

νοσοκομειακό περιβάλλον, αλλά έχουν ένα ευρύτερο φάσμα εφαρμογών. Γι' αυτό το λόγο μια πληροφορία για έναν ασθενή πρέπει να είναι ταυτοποιημένη για να γίνει αποδεκτή από το μεγάλο εύρος απαιτήσεων των διαφορετικών συστημάτων.

*Μερικά παραδείγματα μηνυμάτων είναι:*

- **RQI/RPI - REQUEST FOR INSURANCE INFORMATION (EVENT I01):** Αίτηση Πληροφορίας Ασφαλιστικής Κάλυψης.
- **RQI/RPL - REQUEST/RECEIPT OF PATIENT SELECTION DISPLAY (EVENT I03):** Αίτηση Λίστας Επιλογής Ασθενούς.
- **RQC/RCI - REQUEST FOR PATIENT CLINICAL INFORMATION (EVENT I05):** Αίτηση Κλινικών Πληροφοριών.
- **RQA/RPA - REQUEST PATIENT AUTHORIZATION MESSAGE (EVENT I08):** Αίτηση Μηνύματος Έγκρισης Ασθενούς.
- **RQA/RPA - REQUEST FOR TREATMENT AUTHORIZATION INFORMATION (EVENT I08):** Αίτηση για Έγκριση Θεραπείας.

#### **5.17.11. CLINICAL LABORATORY AUTOMATION – ΑΥΤΟΜΑΤΙΣΜΟΣ ΚΛΙΝΙΚΟΥ ΕΡΓΑΣΤΗΡΙΟΥ**

*Ο* αυτοματισμός ενός κλινικού εργαστηρίου είναι άμεσα συνδεδεμένος με τις διεπαφές αυτόματων ρομποτικών συστημάτων μεταφοράς, αναλυτικών οργάνων και άλλων εργαστηριακών εργαλείων. Μαζί με τις ηλεκτρονικές ή μηχανικές διεπαφές, οι υπολογιστές που ελέγχουν αυτές τις συσκευές πρέπει και να επικοινωνούν μεταξύ τους.

Οι τύποι πληροφοριών που πρέπει να ανταλλάσσονται μέσα σε ένα αυτόματο εργαστήριο είναι πολλοί και διαφορετικοί, όπως έλεγχος της κατάστασης και διαδικασίας για κάθε συσκευή και πληροφορίες για κάθε δείγμα. Επίσης τα αποτελέσματα από τις αναλύσεις πρέπει να διοχετεύονται στη κεντρική βάση δεδομένων του εργαστηρίου. Χωρίς αυτές τις πληροφορίες είναι αδύνατο να λειτουργήσει ένα αυτοματοποιημένο εργαστήριο. Παραδείγματα:

- **ESU/ACK - AUTOMATED EQUIPMENT STATUS UPDATE (EVENT U01):** Ενημέρωση κατάστασης αυτομάτου εξοπλισμού.
- **ESR/ACK - AUTOMATED EQUIPMENT STATUS REQUEST (EVENT U02):** Αίτηση ενημέρωσης αυτομάτου εξοπλισμού.

#### **5.17.12. PERSONNEL MANAGEMENT – ΔΙΑΧΕΙΡΙΣΗ ΠΡΟΣΩΠΙΚΟΥ**

Οι συναλλαγές Διαχείρισης Προσωπικού καθορίζουν τους κανόνες μεταφοράς νέων ή ενημερωμένων πληροφοριών διαχείρισης που αφορούν στους γιατρούς ή στο προσωπικό. Οι συναλλαγές Διαχείρισης Προσωπικού χρησιμοποιούνται προκειμένου να διευκολύνεται η λειτουργία πολλών συστημάτων που σχετίζονται με το προσωπικό, όπως συστήματα ασφάλειας, προγραμματισμού, παραγγελιών κτλ. Για παράδειγμα, είναι σημαντικό για ένα σύστημα ασφαλείας να γνωρίζει πότε ένα μέλος του προσωπικού έχει προσληφθεί. Σε γενικές γραμμές οι πληροφορίες εισέρχονται στο σύστημα διαχείρισης προσωπικού και περνούν από εκεί σε άλλα συστήματα που ζητούν πληροφορίες για το προσωπικό είτε σαν αυτόκλητη ενημέρωση είτε σαν απάντηση σε μία αίτηση.

Μερικά από τα μηνύματα που ανταλλάσσονται στη διαχείριση προσωπικού είναι:

- **PMU/ACK – ADD PERSONNEL RECORD (EVENT B01):**  
Εισαγωγή φακέλου προσωπικού.
- **MU/ACK – UPDATE PERSONNEL RECORD (EVENT B02):**  
Ενημέρωση φακέλου προσωπικού.
- **PMU/ACK – DELETE PERSONNEL RECORD (EVENT B03):** Διαγραφή φακέλου προσωπικού.
- **QBP/RSP – QUERY INFORMATION (EVENT Q25/K25):**  
Αίτηση πληροφοριών.

#### **5.17.13. PATIENT CARE – ΠΕΡΙΘΑΛΨΗ ΑΣΘΕΝΟΥΣ**

Αυτή η εφαρμογή σχετίζεται με ιατρικά μηνύματα που πρέπει να ανταλλάσσονται μεταξύ κλινικών εφαρμογών για κάποιο άτομο. Οι συναλλαγές αυτές μπορούν να γίνουν συνολικά (batch) ή online. Παραδείγματα:

- **QRY - PATIENT CARE PROBLEM QUERY (EVENT PC4):**  
Αίτηση προβλήματος περίθαλψης ασθενούς.
- **PRR - PATIENT PROBLEM RESPONSE (EVENT PC5):**  
Απάντηση σε πρόβλημα ασθενούς.
- **QRY - PATIENT GOAL QUERY (EVENT PC9):** Αίτηση στόχου ασθενούς.
- **PPV - PATIENT GOAL RESPONSE (EVENT PCA):**  
Απάντηση στόχου ασθενούς.



## ΚΕΦΑΛΑΙΟ 6 : ΕΞΕΛΙΞΗ ΤΟΥ ΠΡΟΤΥΠΟΥ

### HL7 VERSION 3

*Η* έκδοση 3 αποτελεί έναν επαναπροσδιορισμό του προτύπου HL7, στην προσπάθεια του Οργανισμού να προσπελάσει μερικά από τα προβλήματα που προέκυψαν από τις εκδόσεις 2.x.

*Από* την Άνοιξη του 1997, οι τεχνικές επιτροπές του HL7 κατευθύνονται στην υλοποίηση της έκδοσης 3, η οποία χρησιμοποιεί μεθοδολογία βασισμένη στο Reference Information Model (Μοντέλο Αναφοράς Πληροφοριών).

*Στόχο* του HL7 αποτελεί η υλοποίηση ενός προτύπου επαρκώς προσδιορισμένου και δοκιμασμένου, το οποίο θα παρέχει την δυνατότητα πιστοποίησης. Για την επίτευξη του στόχου αυτού χρησιμοποιούνται αυστηρές και αναλυτικές τεχνικές κατασκευής μηνυμάτων. Έτσι ενσωματώνονται περισσότερα trigger event και περισσότεροι τύποι μηνυμάτων με πολύ λιγότερη προαιρετικότητα. Αλλάζει το περιεχόμενο μηνυμάτων και πεδίων, όπως και οι κανόνες κωδικοποίησης, τα LLP (low level communication protocols) και οι βασικοί τύποι δεδομένων.

*Η* έκδοση 3 είναι αντικειμενοστραφής και χρησιμοποιεί αποκλειστικά την κωδικοποίηση XML4 για την ανταλλαγή των δεδομένων αντί του απλού κειμένου ASCII που χρησιμοποιούν οι εκδόσεις 2.x.

## **HL7 VERSION 2.5**

*Η* έκδοση 2.5 περιέχει έναν αριθμό νέων events, segments και messages. Αναγνωρίστηκε ως ANSI πρότυπο στις 26 Ιουνίου 2003. Η έκδοση 2.5 είναι περισσότερο λειτουργική από οποιαδήποτε από τις προηγούμενες εκδόσεις.

*Οι* τροποποιήσεις που έγιναν σε σχέση με την έκδοση 2.4 περιλαμβάνουν :

- Βελτιωμένη τεκμηρίωση των τύπων των δεδομένων.
- Τον ορισμό της μεθοδολογίας του message profile.
- Καλύτερη απεικόνιση με την βοήθεια ενός νέου segment και μίας νέας ακολουθίας μηνυμάτων.
- Υποστήριξη παραγγελιών που αφορούν σε παράγωγα αίματος.
- Ένα νέο μήνυμα το οποίο υποστηρίζει διαδικασία διάγνωσης με “update” μέθοδο.
- Μια νέα προδιαγραφή που αφορά σε μηνύματα σχετικά με αποζημιώσεις.

## **HL7 VERSION 2.4**

*Η* έκδοση 2.4 εισάγει το Conformance Query Profiles στο 5ο κεφάλαιο και προσθέτει μηνύματα για την αυτοματοποίηση των εργαστηρίων, την διαχείριση των εφαρμογών και την διαχείριση του προσωπικού. Αναγνωρίστηκε ως ANSI πρότυπο στις 6 Οκτωβρίου 2000. Επιπλέον, προσθέτει ένα νέο event, σχετικό με αποζημιώσεις.

*Τα* κεφάλαια που περιλαμβάνει η έκδοση 2.4 είναι:

- Introduction: Γενική επισκόπηση του HL7.

- Control: Ορισμός Μηνυμάτων, Πρωτόκολλα Ανταλλαγής.
- Patient Administration: Εισαγωγή, Έξοδος, Μεταφορά και δημογραφικά του ασθενούς.
- Order entry: Παραγγελίες για Κλινικές Υπηρεσίες και Παρατηρήσεις, Φαρμακείο, δίαιτα και Προμήθειες.
- Query: Κανόνες που εφαρμόζονται σε Ερωτήματα και στις Απαντήσεις τους.
- Financial Management: Χρεώσεις Ασθενών και Οικονομική διαχείριση.
- Observation Reporting: Μηνύματα Αναφοράς Παρατηρήσεων.
- Master Files: Κύρια Αρχεία Εφαρμογών Φροντίδας Υγείας.
- Medical Records/Information: Διαχείριση εγγράφων, υπηρεσιών Management και πόρων.
- Scheduling: Προγραμματισμένα Ραντεβού.
- Patient Referral :Μηνύματα Παραπομπής Πρωτοβάθμιας Φροντίδας.
- Patient Care: Εγγραφές προσανατολισμένες στο πρόβλημα.
- Laboratory Automation: Κατάσταση εργαστηριακού εξοπλισμού, κατάσταση των δειγμάτων, κατάλογος του εξοπλισμού, ρυθμίσεις του εξοπλισμού, service του εξοπλισμού.
- Application Management: Αιτήματα εφαρμογής στο επίπεδο ελέγχου, μετάδοση διοικητικών πληροφοριών εφαρμογής.
- Personnel Management: Εκπαιδευτικές λεπτομέρειες, λεπτομέρειες γλώσσας, μονάδα οργάνωσης των επαγγελματιών, ταυτοποίηση του προσωπικού.
- Appendix A- Data Definition Tables: Όλοι οι Πίνακες (user defined) του HL7 και οι τιμές τους.
- Appendix B- Lower Layer Protocols: Πρωτόκολλα για τα χαμηλά στρώματα του μοντέλου OSI.
- Appendix C- BNF Message: Αναπαραστάσεις των μηνυμάτων σε Description γλώσσα προγραμματισμού BNF.
- Appendix E- Glossary: Λεξιλόγιο.

## **HL7 VERSION 2.3.1**

*Η έκδοση 2.3.1 αναγνωρίστηκε ως ANSI πρότυπο στις 14 Απριλίου 1999. Περιλαμβάνει ένα αναβαθμισμένο T/Q (timing/quantity) είδος δεδομένων προκειμένου να διαχειριστεί την εμφάνιση των παραγγελιών, τις αναβαθμίσεις στο OBR segment, στο ORU message, σε πίνακες, segment και σε είδη δεδομένων προκειμένου να προσαρμοστούν σε διεθνείς υποδείξεις. Επιπλέον περιέχει ένα νέο πεδίο στο segment ORC για τις απαιτήσεις εξωτερικών ασθενών.*

*Τα κεφάλαια που περιλαμβάνει η έκδοση 2.4 είναι:*

- Introduction: Γενική επισκόπηση του HL7.
- Control: Ορισμός Μηνυμάτων, Πρωτόκολλα Ανταλλαγής.
- Patient Administration: Εισαγωγή, Έξοδος, Μεταφορά και δημογραφικά του ασθενούς.
- Order entry: Παραγγελίες για Κλινικές Υπηρεσίες και Παρατηρήσεις, Φαρμακείο, δίαιτα και Προμήθειες.
- Query: Κανόνες που εφαρμόζονται σε Ερωτήματα και στις Απαντήσεις τους.
- Financial Management: Χρεώσεις Ασθενών και Οικονομική διαχείριση.
- Observation Reporting: Μηνύματα Αναφοράς Παρατηρήσεων.
- Master Files: Κύρια Αρχεία Εφαρμογών Φροντίδας Υγείας.
- Medical Records/ Information: Διαχείριση Εγγράφων, Υπηρεσιών Management και Πόρων.
- Scheduling: Προγραμματισμένα Ραντεβού.
- Patient Referral: Μηνύματα Παραπομπής Πρωτοβάθμιας Φροντίδας.
- Patient Care: Εγγραφές προσανατολισμένες στο πρόβλημα.
- Appendix A- Data Definition Tables: Όλοι οι Πίνακες (user defined) του HL7 και οι τιμές τους.

- Appendix B- Lower Layer Protocols: Πρωτόκολλα για τα χαμηλά στρώματα του μοντέλου OSI.
- Appendix C- Network Management: Αιτήματα εφαρμογής στο επίπεδο ελέγχου, μετάδοση διοικητικών πληροφοριών εφαρμογής.
- Appendix C- BNF Message: Αναπαραστάσεις των μηνυμάτων σε Description γλώσσα προγραμματισμού BNF.
- Appendix E- Glossary: Λεξιλόγιο.

### **HL7 VERSION 2.3**

*Η* έκδοση 2.3 αναγνωρίστηκε ως ANSI πρότυπο στις 13 Μαΐου 1997. Περιλαμβάνει ένα αναβαθμισμένο T/Q (timing/quantity) είδος δεδομένων προκειμένου να διαχειριστεί την εμφάνιση των παραγγελιών, τις αναβαθμίσεις στο OBR segment, στο ORU message, σε πίνακες, σε segment και σε είδη δεδομένων προκειμένου να προσαρμοστούν σε διεθνείς υποδείξεις.

*Η* έκδοση 2.3 περιέχει μηνύματα διαχείρισης εγγράφων, μηνύματα για υπηρεσίες ραντεβού και για σχεδιασμό πόρων, μηνύματα για παραπομπές ασθενών και για την ικανοποίηση αιτημάτων αυτών.

*Η* δομή της έκδοσης 2.3 είναι η παρακάτω.

- Introduction: Γενική επισκόπηση του HL7.
- Control: Ορισμός Μηνυμάτων, Πρωτόκολλα Ανταλλαγής.
- Patient Administration: Εισαγωγή, Έξοδος, Μεταφορά και δημογραφικά του ασθενούς.
- Order entry: Παραγγελίες για Κλινικές Υπηρεσίες και Παρατηρήσεις, Φαρμακείο, δίαιτα και Προμήθειες.

- Query: Κανόνες που εφαρμόζονται σε Ερωτήματα και στις Απαντήσεις τους.
- Financial Management : Χρεώσεις Ασθενών και Οικονομική διαχείριση.
- Observation Reporting: Μηνύματα Αναφοράς Παρατηρήσεων.
- Master Files: Κύρια Αρχεία Εφαρμογών Φροντίδας Υγείας.
- Medical Records/Information: Διαχείριση Εγγράφων, Υπηρεσιών Management και Πόρων.
- Scheduling: Προγραμματισμένα Ραντεβού.
- Patient Referral: Μηνύματα Παραπομπής Πρωτοβάθμιας Φροντίδας.
- Patient Care: Εγγραφές προσανατολισμένες στο πρόβλημα.
- Appendix A- Data Definition Tables: Όλοι οι Πίνακες (user defined) του HL7 και οι τιμές τους.
- Appendix B- Lower Layer Protocols: Πρωτόκολλα για τα χαμηλά στρώματα του μοντέλου OSI.
- Appendix C- Network Management: Αιτήματα εφαρμογής στο επίπεδο ελέγχου, μετάδοση διοικητικών πληροφοριών εφαρμογής.
- Appendix C- BNF Message: Αναπαραστάσεις των μηνυμάτων σε Descriptions γλώσσα προγραμματισμού BNF.
- Appendix E- Glossary: Λεξιλόγιο.

## ΠΛΕΟΝΕΚΤΗΜΑΤΑ ΤΩΝ ΕΚΔΟΣΕΩΝ 2.x ΤΟΥ HL7

Τα μηνύματα των εκδόσεων 2.x. εφαρμόστηκαν ευρέως σε διάφορες εφαρμογές σε ποικίλα περιβάλλοντα υγειονομικής περίθαλψης και συχνά με επιτυχία. Η διάδοσή τους οφείλεται στα βασικά πλεονεκτήματά τους, την **ευελιξία (flexibility)** και την **προαιρετικότητα (optionality)**. Οι εκδόσεις αυτές περιλαμβάνουν πολλά προαιρετικά πεδία με συνέπεια να γίνονται αποδεκτές σε

οποιαδήποτε εφαρμογή. Αυτό σημαίνει εύκολη προσαρμογή στις απαιτήσεις υλοποίησης HL7 μηνυμάτων σε μία Μονάδα Υγείας.

### **ΜΕΙΟΝΕΚΤΗΜΑΤΑ ΤΩΝ ΕΚΔΟΣΕΩΝ 2.x ΤΟΥ HL7**

Σε αντίθεση με άλλα διεθνή πρωτόκολλα επικοινωνίας, το HL7 δεν είναι plug and play, απαιτούνται δηλαδή, μετά την εγκατάσταση, ειδικές επεμβάσεις και μετατροπές στον συντακτικό αναλυτή (parser) του HL7 καθώς και στις ίδιες τις εφαρμογές προτού ξεκινήσει η λειτουργία του συστήματος. Το γεγονός αυτό οφείλεται στο ευρύ φάσμα των εφαρμογών που καλείται να διασυνδέσει το HL7.

Συχνά κατασκευαστές λογισμικού υποστηρίζουν ότι τα συστήματά τους είναι συμβατά με κάποια έκδοση του προτύπου χωρίς όμως να παρέχουν καμιά τεκμηρίωση των μηνυμάτων που υποστηρίζουν. Δεν εξασφαλίζεται, εξαιτίας της προαιρετικότητας, η διαλειτουργικότητα μεταξύ συστημάτων διαφορετικών προμηθευτών. Έτσι:

- Λείπουν ολόκληρα πεδία. Αντί για να αφήνουν κενό ένα πεδίο, ορισμένοι προμηθευτές το απαλείφουν εντελώς.
- Τα ίδια δεδομένα παρουσιάζονται σε διαφορετικά πεδία.
- Τα ίδια δεδομένα παρουσιάζονται με διαφορετική μορφοποίηση (format).
- Η ύπαρξη διαφορετικών εκδόσεων (versions) του HL7 καθιστά δυνατή την ανταλλαγή δεδομένων μόνο μεταξύ εφαρμογών που υποστηρίζουν την ίδια έκδοση.
- Λείπουν τιμές (values) ακόμα και από υποχρεωτικά πεδία. Παρά το γεγονός ότι το πρωτόκολλο απαιτεί ένα minimum αριθμό υποχρεωτικών πεδίων (το 95% είναι προαιρετικά), ορισμένοι προμηθευτές λογισμικού παραβαίνουν αυτόν τον κανόνα.

- Μη ορθή σύνταξη του μηνύματος. Ορισμένα αναμενόμενα τμήματα απουσιάζουν ή εμφανίζονται άλλα, μη αναμενόμενα.
- Απαιτούνται ελεγχόμενα λεξικά.
- Μη σαφής υποστήριξη για νέες τεχνολογίες: Object τεχνολογίες, XML και Web τεχνολογίες καθώς και ελλιπής υποστήριξη ασφαλείας.

Όλοι αυτοί οι περιορισμοί καθιστούν πολύπλοκη την ολοκλήρωση ενός συστήματος βασισμένου στην έκδοση 2.χ του HL7, με αποτέλεσμα να απαιτείται ένα χρονικό διάστημα τουλάχιστον 2-4 μηνών για την εγκατάσταση των διεπαφών του HL7.



## ΚΕΦΑΛΑΙΟ 7: ΠΕΙΡΑΜΑΤΙΚΗ ΕΦΑΡΜΟΓΗ

### 7.1. ΧΑΡΑΚΤΗΡΙΣΤΙΚΑ

Στην παρούσα εργασία προσπαθήσαμε να υλοποιήσουμε μία εφαρμογή προκειμένου να επιτευχθεί η επικοινωνία ανάμεσα σε δύο ιατρικά πληροφοριακά συστήματα, τα οποία ενδεχομένως είναι ετερογενή και απομακρυσμένα.

Ο τρόπος που επιλέξαμε να το κάνουμε αυτό είναι η χρησιμοποίηση της XML τεχνολογίας. Πιο συγκεκριμένα, προκειμένου να μεταφερθεί ένα HL7 μήνυμα μέσω πολλαπλών πληροφοριακών συστημάτων, έπρεπε να μπορεί να γίνει κατανοητό και από τα δύο συνδιαλλεγόμενα μέρη. Ένας εύκολος τρόπος για να γίνει αυτό είναι η μετατροπή του μηνύματος από HL7 κώδικα σε XML μορφή.

Σκοπός λοιπόν της πειραματικής μας υλοποίησης, είναι η ανάπτυξη μιας εφαρμογής που θα επιτρέπει τη μετατροπή των μηνυμάτων από την HL7 μορφή τους σε XML. Γι αυτό το σκοπό χρησιμοποιήσαμε τη γλώσσα προγραμματισμού PHP. Η γλώσσα αυτή είναι αρκετά απλή και παρουσιάζει αρκετές ομοιότητες με την HTML, αλλά σίγουρα έχει περισσότερες δυνατότητες.

Μερικά από τα πλεονεκτήματα της PHP είναι:

1) Η ύπαρξη αντικειμενοστρέφειας, 2) Η διαδικασία διασύνδεσης με μια Βάση Δεδομένων είναι απλή, 3) Η σύνταξη είναι όμοια με της C και επομένως είναι πιο εύκολο να προσαρμοστεί ένας προγραμματιστής και να τη μάθει γρήγορα, 4) Ο εμπλουτισμός της λειτουργικότητας των ιστοσελίδων, 5) Η απλή σύνταξη, 6) Είναι

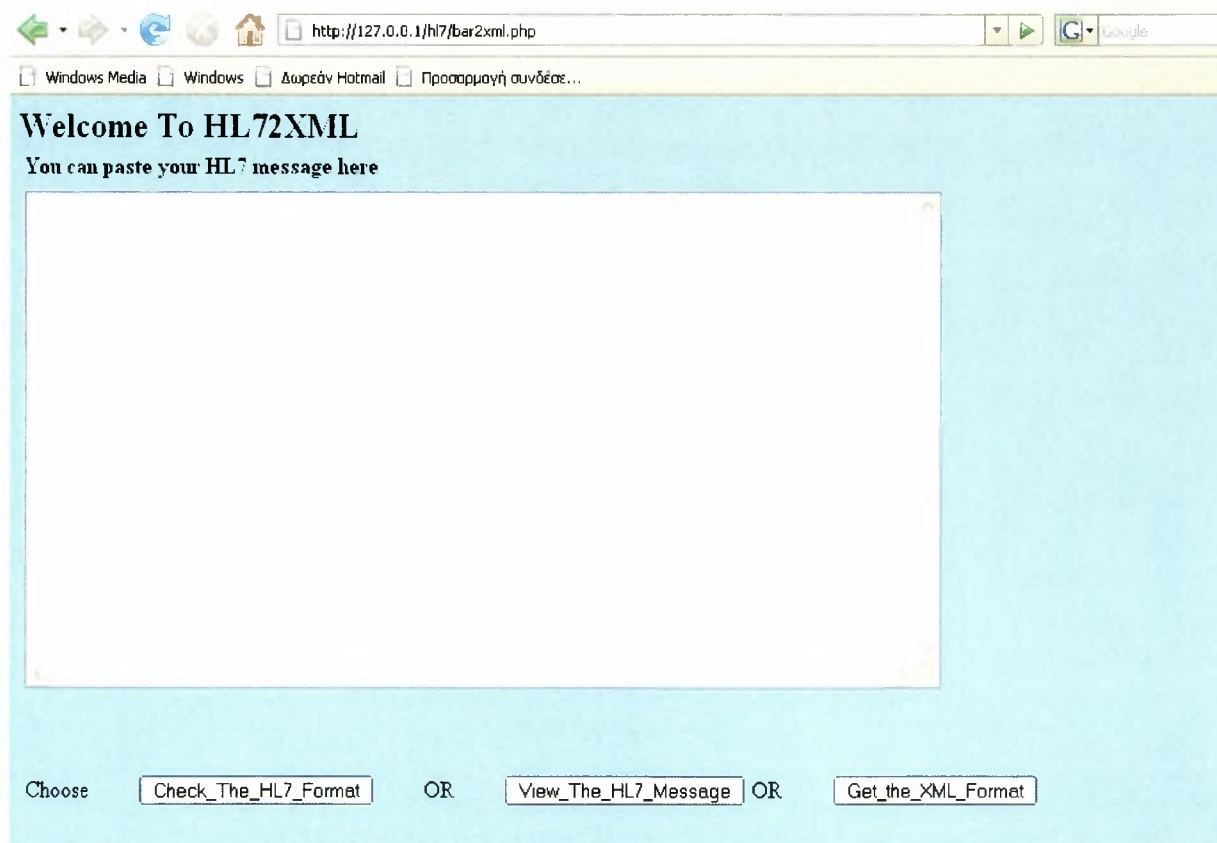
αρκετά διαδεδομένη και υποστηρίζεται από μεγάλη κοινότητα χρηστών και προγραμματιστών

## **7.2. ΤΕΧΝΟΛΟΓΙΚΟ ΠΕΡΙΒΑΛΛΟΝ**

Στην παράγραφο αυτή θα αναφερθούμε στις τεχνολογίες που χρησιμοποιήσαμε για να αναπτύξουμε την εφαρμογή μας. Προκειμένου να μπορέσουμε να εκτελέσουμε τον κώδικα της PHP, χρησιμοποιήσαμε το λειτουργικό σύστημα των Windows XP και την πλατφόρμα EASYPHP 2.0 b1, η οποία περιλαμβάνει την εγκατάσταση ενός HTTP Server, του Apache 1.3.33 (έκδοση Φεβρουάριος 2002), της php 4.3.10, της mysql 4.1.9 και της phpmyadmin 2.6.1. Αναπτύξαμε σε php, το αρχείο hl7\_to\_xml.php, το οποίο μετατρέπει τα HL7 μηνύματα στην HL7 Version2.xml μορφή τους, με τη χρήση έτοιμων συναρτήσεων και parser. Προκειμένου να γίνει η μετατροπή, το πρόγραμμά μας χρησιμοποιεί τα HL7 XML Schema definition files (\*.xsd). Στο παράρτημα παραθέτουμε μόνο τον κώδικα των συναρτήσεων που αναπτύξαμε σε Php, για να λειτουργήσει η εφαρμογή σε έναν επιτραπέζιο ή φορητό υπολογιστή, θα πρέπει να γίνουν οι κατάλληλες προσθήκες και τροποποιήσεις των free – source συναρτήσεων, καθώς και να προσθέσουμε στην αρχή του αρχείου τις σωστές θέσεις αυτών καθώς και των μεταβλητών που χρησιμοποιούν.

Προκειμένου λοιπόν να δούμε πως λειτουργεί η εφαρμογή σε έναν υπολογιστή, αφού κάνουμε τις κατάλληλες προσθήκες που προαναφέραμε, τρέχουμε το αρχείο hl7\_to\_xml.php σαν μία php εφαρμογή μέσω ενός web browser. Ανοίγουμε την πλατφόρμα EASYPHP. Αν έχει εγκατασταθεί σωστά, τότε στη γραμμή εργασιών του υπολογιστή μας (κάτω δεξιά) θα εμφανιστεί το εικονίδιο του προγράμματος αυτού. Στη συνέχεια, ανοίγουμε έναν web browser, τον Internet Explorer ή τον Mozilla Firefox και στη γραμμή

διευθύνσεων συμπληρώνουμε τη διεύθυνση του υπολογιστή μας, δηλαδή γράφουμε: <http://127.0.0.1/>. Αμέσως θα ανοίξει ένα παράθυρο με τα περιεχόμενα της διεύθυνσης 127.0.0.1. Εμείς επιλέγουμε να ανοίξουμε το αρχείο hl7\_to\_xml.php και αμέσως εμφανίζεται το παρακάτω παράθυρο του Internet Explorer:



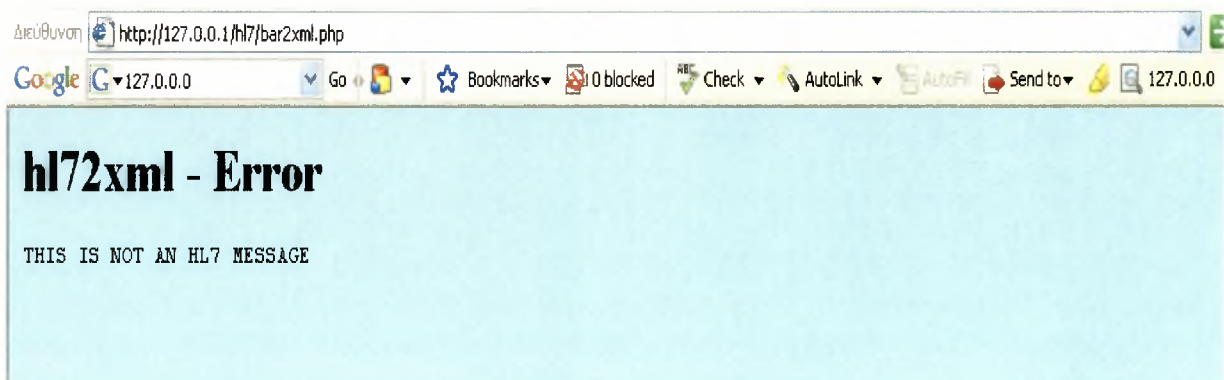
Εικ.1

Στο παράθυρο που άνοιξε, παρατηρούμε ουσιαστικά τέσσερα πράγματα. Αρχικά, βλέπουμε το χώρο που υπάρχει, κάτω από το: ***You can paste your HL7 message here***, στον οποίο μπορούμε να γράψουμε ή να αντιγράψουμε το HL7 μήνυμά μας. Στη συνέχεια, όταν εισάγουμε το μήνυμά μας, υπάρχει η δυνατότητα τριών επιλογών: α) ***Check\_the\_HL7\_Format***: Μέσω της οποίας ελέγχεται εάν το μήνυμα που δώσαμε είναι ένα HL7 μήνυμα και όχι κάτι άλλο. β) ***View\_the\_HL7\_Message***: Με αυτή την επιλογή μπορούμε να δούμε πάλι το μήνυμα που δώσαμε στο πρόγραμμά μας. γ)

**Get\_the\_XML\_Format:** Εδώ μπορούμε να δούμε την XML δομή του μηνύματός μας.

Στη συνέχεια παραθέτουμε εικόνες με όλες τις επιλογές που διαθέτει το πρόγραμμά μας.

Αν στην περιοχή επικόλλησης του HL7 μηνύματος, αντιγράψουμε ένα κείμενο το οποίο δεν είναι HL7 (π.χ. γράψουμε τη λέξη hi), οποιαδήποτε και από τις τρεις επιλογές αν πατήσουμε, τότε θα δούμε την παρακάτω εικόνα:



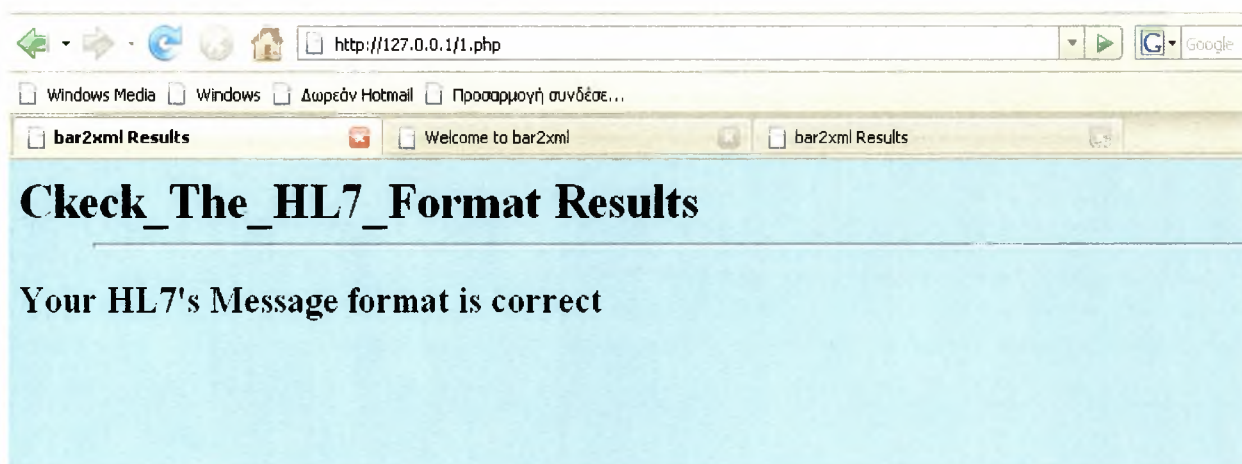
Εικ.2

Όταν αντιγράψουμε το παρακάτω HL7 μήνυμα:

```
MSH|^~\&||ABCHS||AUSDHSV|20070101112951||ADT^A04^ADT_A  
01|12334456778890|P|2.5||NE|NE|AU|ASCII  
EVN|A04|20060705000000  
PID|1||0000112234^^^100^A||XXXXXXXXXX^^^^^S||10131113|1||  
4|^^RICHMOND^^3121|||1201|||||1100|||||AAA  
PD1||2  
NK1|1|1|||||||||||2  
PV1|1|O|||^^^^1
```

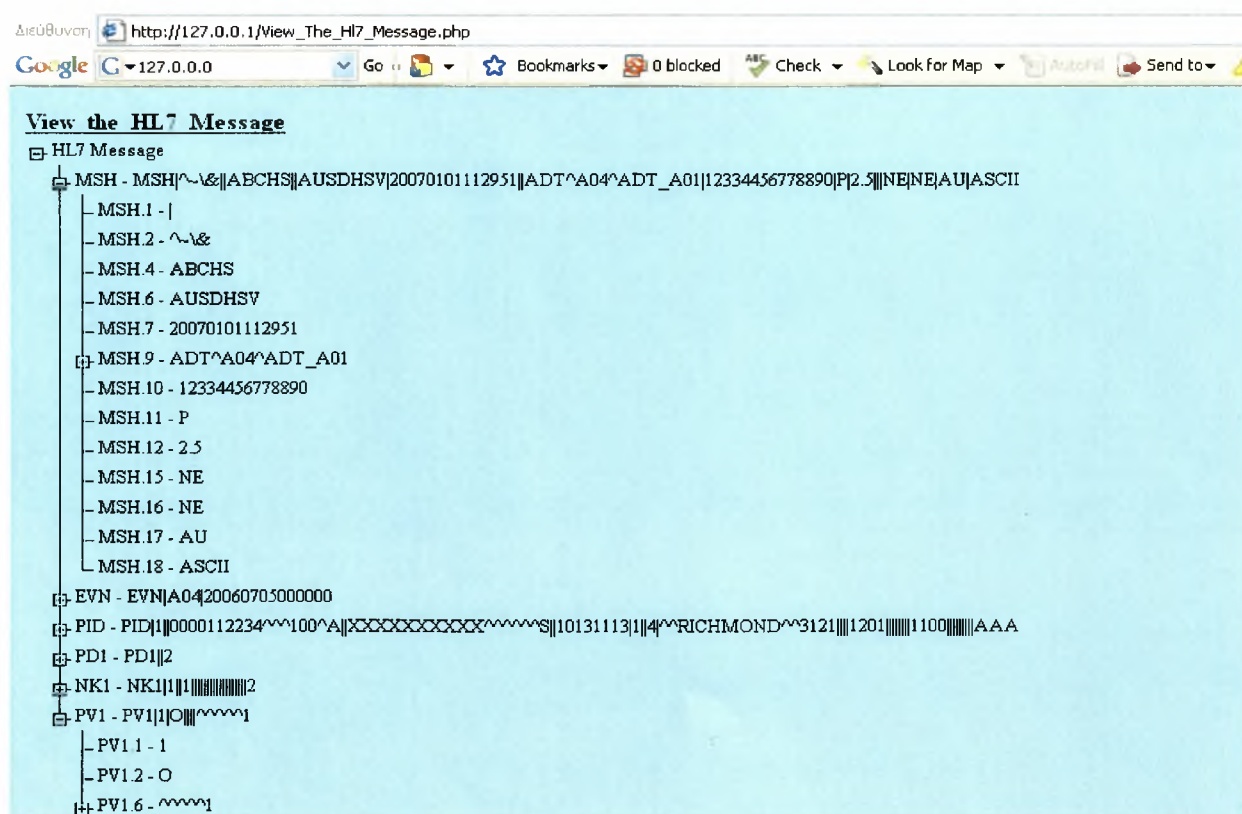
τότε για κάθε μία από τις τρεις επιλογές που υπάρχουν θα δούμε τα παρακάτω παράθυρα:

Πατάμε: **Check\_The\_HL7\_Format** και βλέπουμε ένα τέτοιο παράθυρο, το οποίο μας ενημερώνει ότι η δομή του μηνύματος είναι σωστή:



Εικ.3

Πατάμε: **View\_the\_HL7\_Message** και βλέπουμε το παρακάτω παράθυρο, που μας δείχνει τα πεδία του μηνύματός μας:



#### Εικ.4

Πατάμε: **Get\_the\_XML\_Format** και εμφανίζεται το παράθυρο που μας δίνει την XML μορφή του μηνύματός μας:

```
<?xml version="1.0" encoding="UTF-8" ?>
<ADT_A01 xmlns="urn:hl7-org:v2xml"
  xmlns:xsi="http://www3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="urn:hl7-org:v2xml ADT_A01.xsd">
<MSH>
  <MSH.1>|</MSH.1>
  <MSH.2>^~\&</MSH.2>
  <MSH.4>ABCHS</MSH.4>
  <MSH.6>AUSDHSV</MSH.6>
  <MSH.7>20070101112951</MSH.7>
<MSH.9>
  <MSG.1>ADT</MSG.1>
  <MSG.2>A04</MSG.2>
  <MSG.3>ADT_A01</MSG.3>
  </MSH.9>
  <MSH.10>12334456778890</MSH.10>
  <MSH.11>P</MSH.11>
  <MSH.12>2.5</MSH.12>
  <MSH.15>NE</MSH.15>
  <MSH.16>NE</MSH.16>
  <MSH.17>AU</MSH.17>
  <MSH.18>ASCII</MSH.18>
  </MSH>
<EVN>
  <EVN.1>A04</EVN.1>
  <EVN.2>20060705000000</EVN.2>
  </EVN>
<PID>
  <PID.1>1</PID.1>
<PID.2>
  <CX.1>0000112234</CX.1>
  <CX.2>100</CX.2>
  <CX.3>A</CX.3>
  </PID.2>
<PID.3>
  <XPN.1>XXXXXXXXXX</XPN.1>
  <XPN.2>S</XPN.2>
  </PID.3>
  <PID.4>10131113</PID.4>
  <PID.5>1</PID.5>
  <PID.6>4</PID.6>
<PID.7>
  <XAD.1>RICHMOND</XAD.1>
  <XAD.1>3121</XAD.2>
```



```

    </PID.7>
    <PID.8>1201</PID.8>
    <PID.9>1100</PID.9>
    <PID.10>AAA</PID.10>
    </PID>
  <PD1>
    <PD1.2>2</PD1.2>
    </PD1>
  <NK1>
    <NK1.1>1</NK1.1>
    <NK1.2>1</NK1.2>
    <NK1.3>2</NK1.3>
    </NK1>
  <PV1>
    <PV1.1>1</PV1.1>
    <PV1.2>0</PV1.2>
  <PV1.3>
    <PL.6>1</PL.6>
    </PV1.3>
  </PV1>
</ADT_A01>

```

### **7.3. ΠΕΡΙΒΑΛΛΟΝ ΛΕΙΤΟΥΡΓΙΑΣ**

Πριν ξεκινήσει η μετατροπή του μηνύματος στην XML μορφή του, ελέγχεται η δομή του μηνύματος, τα κομμάτια (segments) αυτού, οι τύποι δεδομένων (data types) και το μήκος των πεδίων (field length). Στη συνέχεια ελέγχει για το αν το μήνυμα είναι συντακτικά σωστό (δηλαδή, εξασφαλίζεται η ύπαρξη των απαραίτητων κομματιών και πεδίων και η σωστή σειρά εμφάνισής τους. Εάν το μήνυμα αποτύχει στους παραπάνω ελέγχους, τότε το πρόγραμμα μας δε θα εμφανίσει καμία XML μορφή του μηνύματος, εν αντιθέσει θα εμφανιστεί ένα μήνυμα λάθους. Όταν οι παραπάνω έλεγχοι τελειώσουν με επιτυχία, τότε θα ξεκινήσει η διαδικασία μετατροπής του μηνύματος. Κατά τη διάρκεια αυτής επαληθεύεται η ύπαρξη απαραίτητων πεδίων για το μήνυμα, η ύπαρξη επαναλαμβανόμενων ή μη χαρακτήρων, η σωστή χρησιμοποίηση αριθμών και πολλά άλλα. Όλοι οι έλεγχοι που πραγματοποιούνται γίνονται με βάση τους πίνακες που έχει δημοσιεύσει ο οργανισμός HL7.

Με την PHP αναπτύξαμε όλο το γραφικό περιβάλλον μέσω του οποίου ο χρήστης θα μπορεί να έχει πρόσβαση στις επιλογές του προγράμματος.

Πιο συγκεκριμένα, υλοποιήσαμε τις εξής συνάρτησεις:

- `Get_hl7()`, μέσω της οποίας αναπτύξαμε το αρχικό γραφικό περιβάλλον που βλέπει ο χρήστης στην εικόνα 1, δηλαδή το αρχικό παράθυρο.
- `Message_correct()`, μέσω της οποίας ανάπτυξαμε το γραφικό περιβάλλον που βλέπει ο χρήστης στην εικόνα 3, δηλαδή το παράθυρο που βλέπει ο χρήστης όταν το μήνυμά του περνάει τον έλεγχο ορθότητας.
- `hl7xml_destroy ($message)`, η οποία μας εμφανίζει σε παράθυρο, όπως στην εικόνα 2, ότι παρουσιάστηκε ένα σφάλμα κατά την επεξεργασία του μηνύματός μας και καλείται στην `Process()`.
- `Process()`, ελέγχει εάν το μήκος του μηνύματος είναι μεγαλύτερο από 0 χαρακτήρες (εάν είναι μικρότερο εμφανίζει μήνυμα λάθους), εάν είναι μικρότερο από 655360 (εάν είναι μεγαλύτερο εμφανίζει μήνυμα λάθους), αναζητά το MSH segment (εάν δεν το βρει ή εάν έχει μήκος μικρότερο από 12 χαρακτήρες εμφανίζει μήνυμα λάθους), ελέγχει την εγκυρότητα των χαρακτήρων που διαβάζει, κοιτάζει εάν το HL7 μήνυμα ανήκει σε έκδοση που μπορεί να επεξεργαστεί (εάν όχι εμφανίζει ανάλογο μήνυμα), ελέγχει τη δομή του μηνύματος με το να δει εάν λείπουν ορισμένα components και μετατρέπει το μήνυμα στην XML δομή του.

Οι παραπάνω συναρτήσεις κάνουν χρήση open – source συναρτήσεων σε php. Ενδεικτικά θα αναφέρουμε : (1) την `ParseXSD($file)`, κι αυτό γιατί στο παράρτημα με τον κώδικα των παραπάνω θα χρησιμοποιηθεί. Πρόκειται για έναν parser, ο οποίος



είναι υπεύθυνος για τη μετατροπή του μηνύματος στην XML μορφή του και καλείται μέσα στη συνάρτηση `Process()`. Όπου `file`, αρχείο XSD, μέσω του οποίου θα γίνει η μετατροπή. Η συνάρτηση αυτή καλείται μετά την προσπάθεια ανοίγματος του μηνύματος και αφού αυτό έχει περάσει τους ελέγχους που αναφέραμε στην προηγούμενη παράγραφο. Η κλήση της μέσα στην `Process()` γίνεται ως εξής: α) `ParseXSD($XSDdir . 'segments.xsd');` -> Διαβάζει και μετατρέπει σε XML τα κομμάτια του μηνύματος, β) `ParseXSD($XSDdir . 'fields.xsd');` -> Διαβάζει και μετατρέπει τα Field του μηνύματος σε XML, γ) `ParseXSD($XSDdir . 'datatypes.xsd')` -> Διαβάζει και μετατρέπει τους τύπους δεδομένων σε XML μορφή. (2) την `ParseHL7()` με τη χρήση αυτής της συνάρτησης ελέγχουμε αν τα κομμάτια του μηνύματος είναι στη σωστή σειρά, προκειμένου να κατασκευαστεί το XML δέντρο. Η συνάρτηση επιστρέφει τρεις τιμές: α) αριθμός των κομματιών που πέρασαν επιτυχώς τη διαδικασία επεξεργασίας, β) -1, εάν απέτυχε η επεξεργασία ενός κομματιού και γ) -2, εάν προέκυψε πρόβλημα επεξεργασίας όλων των κομματιών του μηνύματος. (3) την `ParseFail ($message, $segNo, $node)`, με τη χρήση αυτής γίνεται ο έλεγχος για το εάν το HL7 μήνυμα μπορεί να μετατραπεί σε XML μέσω της `ParseXSD`, ανάλογα με τις τιμές που επιστρέφει η συνάρτηση `ParseHL7`.

## ΚΕΦΑΛΑΙΟ 8: ΕΠΙΛΟΓΟΣ

### 8.1. ΣΥΜΠΕΡΑΣΜΑΤΑ

Ανακεφαλαιώνοντας λοιπόν, στη διπλωματική μας εργασία είχαμε να εξετάσουμε τα παρακάτω:

- Μελέτη του προβλήματος της διαλειτουργικότητας των Ιατρικών Πληροφοριακών Συστημάτων.
- Ανάλυση των λεγόμενων Διαδικτυακών Υπηρεσιών, μέσω των οποίων εδραιώνεται η επικοινωνία ετερογενών και απομακρυσμένων Πληροφοριακών Συστημάτων.
- Παρουσίαση και μελέτη της τεχνολογίας XML (eXtensible Markup Language) καθώς και της σχέσης που υπάρχει ανάμεσα στην τεχνολογία XML και στα πληροφοριακά συστήματα.
- Παρουσίαση και ανάλυση του προτύπου HL7 (Health Level 7).
- Μελέτη μιας εφαρμογής, μέσω της οποίας θα θέλαμε να προτείνουμε έναν τρόπο ανταλλαγής των HL7 μηνυμάτων.

Με την έρευνα που εκπονήσαμε, διαπιστώσαμε ότι η επίτευξη της διαλειτουργικότητας ανάμεσα σε οποιαδήποτε Πληροφοριακά Συστήματα και κυρίως τα Ιατρικά Πληροφοριακά Συστήματα, αποτελεί πολύ μεγάλο κεφάλαιο σε παγκόσμιο ερευνητικό επίπεδο. Επίσης συμπεράναμε ότι αρχικά τα απομακρυσμένα ιατρικά πληροφοριακά συστήματα επικοινωνούσαν διαμέσου πρωτοκόλλων, τα οποία ανήκουν στην κατηγορία λογισμικού middleware και χρησιμοποιούνται μέχρι και σήμερα. Το σημαντικότερο μειονέκτημά τους που τα καθιστά δύσχρηστα είναι ότι δεν υποστηρίζουν όλα τα πληροφοριακά συστήματα αλλά αυτά για τα οποία ο εκάστοτε μηχανικός λογισμικού έχει ανάπτυξει το συγκεκριμένο πρωτόκολλο.

Όπως προαναφέραμε, η διαλειτουργικότητα των Πληροφοριακών Συστημάτων αποτελούσε και αποτελεί ίσως το μεγαλύτερο παράγοντα που καθιστά ένα Πληροφοριακό Σύστημα εύχρηστο και αποτελεσματικό. Με την έρευνά μας είδαμε έναν καινούργιο μηχανισμό εδραίωσης της επικοινωνίας και της ανταλλαγής δεδομένων ανάμεσα σε απομακρυσμένα ετερογενή ιατρικά πληροφοριακά συστήματα (δηλαδή της επίτευξης της διαλειτουργικότητας), τις Διαδικτυακές Υπηρεσίες (Web Services). Είδαμε ότι οι ενέργειες αυτών περιγράφονται με την τεχνολογία XML, ότι αναγνωρίζονται από το URL τους και το περιβάλλον διεπαφής τους, ότι μπορούν να χρησιμοποιηθούν από πολλά διαφορετικά συστήματα, ότι είναι ανεξάρτητες πλατφόρμας και ότι βασίζονται στις τεχνολογίες SOAP (Simple Object Access Protocol), WSDL (Web Services Description Language) και UDDI (Universal Description, Discovery and Integration), τις οποίες περιγράψαμε αναλυτικά στα σχετικά κεφάλαια.

Στη συνέχεια, αφού συμπεράναμε ότι η χρήση των Διαδικτυακών Υπηρεσιών είναι αναγκαία για την επίτευξη της διαλειτουργικότητας, μελετήσαμε την τεχνολογία στην οποία στηρίζονται, την XML. Είδαμε ότι αναπτύχθηκε από το W3C's XML Working Group το 1996, ότι αποτελεί έναν τρόπο αναπαράστασης δεδομένων ανεξάρτητο από συγκεκριμένα συστήματα, ότι είναι μια ευρέως υποστηριζόμενη και ανοικτή τεχνολογία για την περιγραφή δεδομένων, ότι ουσιαστικά πρόκειται για ένα σύνολο κανόνων για το σχεδιασμό μορφών κειμένου οι οποίες διευκολύνουν τη δόμηση των δεδομένων, ότι παρουσιάζει αρκετά πλεονεκτήματα χρήσης συγκριτικά με άλλες τεχνολογίες, ότι έχει συγκεκριμένη δομή και σύνταξη, ότι συνδυάζει αρκετές τεχνολογίες (XSL, XSLT, XML Schemas κλπ) και ότι συνδέεται άμεσα με τα ιατρικά πληροφοριακά συστήματα και πιο συγκεκριμένα με το πρότυπο HL7.

*Η έρευνά μας προχώρησε στη μελέτη του κυριότερου προτύπου των Ιατρικών Πληροφοριακών Συστημάτων, του HL7. Διαπιστώσαμε μέσα σε πολλά άλλα ότι πρόκειται για ένα δομημένο (structured), μηνυματοκεντρικό (message-oriented) πρωτόκολλο που εξασφαλίζει την επικοινωνία απομακρυσμένων υπολογιστών και την παροχή υπηρεσιών ιατρικής περίθαλψης, ότι λειτουργεί στο επίπεδο 7 του επιπέδου OSI, ότι έχει συγκεκριμένη δομή, ότι μπορεί να οριστεί σαν ένα εργαλείο, με ειδική εγγραφή σε κάθε σύστημα, που διευκολύνει διαφορετικά συστήματα ή ιατρικά μηχανήματα, να επικοινωνούν με ευχέρεια μεταξύ τους και να διαβιβάζουν τα δεδομένα που περιέχουν το ένα στο άλλο, ότι έχει συγκεκριμένη σύνταξη, ότι είναι το πλέον ευρέως χρησιμοποιούμενο πρότυπο ανταλλαγής πληροφοριών μέσω μηνυμάτων σε κλινικό περιβάλλον, ότι χρησιμοποιείται σε όλες τις ηπείρους και ότι είναι ένας κώδικας επικοινωνίας, κοινά αποδεκτός από όλους τους κατασκευαστές ιατρικών πληροφοριακών συστημάτων.*

*Παραπάνω αναφέραμε ότι μέσω της έρευνας μας διαπιστώσαμε ότι το πρότυπο HL7 συνδέεται άμεσα με την τεχνολογία XML. Είδαμε ότι πολλές ερευνητικές ομάδες εργασίας ανά τον κόσμο ασχολούνται με τη μετατροπή των HL7 μηνυμάτων σε XML κωδικοποίηση. Ο κυριότερος λόγος για τον οποίο επελέχθη να είναι η XML η γλώσσα αναπαράστασης των HL7 μηνυμάτων είναι ότι παρέχει τη δυνατότητα για μεταφορά και ανταλλαγή των μετασχηματισμένων δεδομένων ανάμεσα σε ετερογενείς πλατφόρμες, αφού όπως είπαμε η τεχνολογία XML είναι ανεξάρτητη πλατφόρμας. Για το σκοπό αυτό παρατηρήσαμε ότι έχουν αναπτυχθεί πολλές πλατφόρμες και προγράμματα σε αρκετές γλώσσες προγραμματισμού (Java, C/C++, Python, Perl, Php κλπ), οι οποίες σκοπό έχουν να δέχονται ένα HL7 μήνυμα ως είσοδο και στην έξοδό τους να παράγουν ένα δένδρο XML που να περιέχει όλα τα δεδομένα που υπήρχαν στην αρχική είσοδο. Η πειραματική μας εφαρμογή στηρίχθηκε στην Php. Τα*

πλεονεκτήματα της γλώσσας αυτής τα εκθέσαμε αναλυτικά παραπάνω και γι αυτό το λόγο θα περιοριστούμε στο να επαναλάβουμε ότι δεν έχει τις δυσκολίες που παρουσιάζονται σε άλλες γλώσσες προγραμματισμού και ότι μπορούμε πολύ εύκολα με τη χρήση ενός web browser να δούμε την εφαρμογή μας χωρίς να είναι απαραίτητη η εγκατάσταση πολύπλοκων προγράμματος στον υπολογιστή μας.

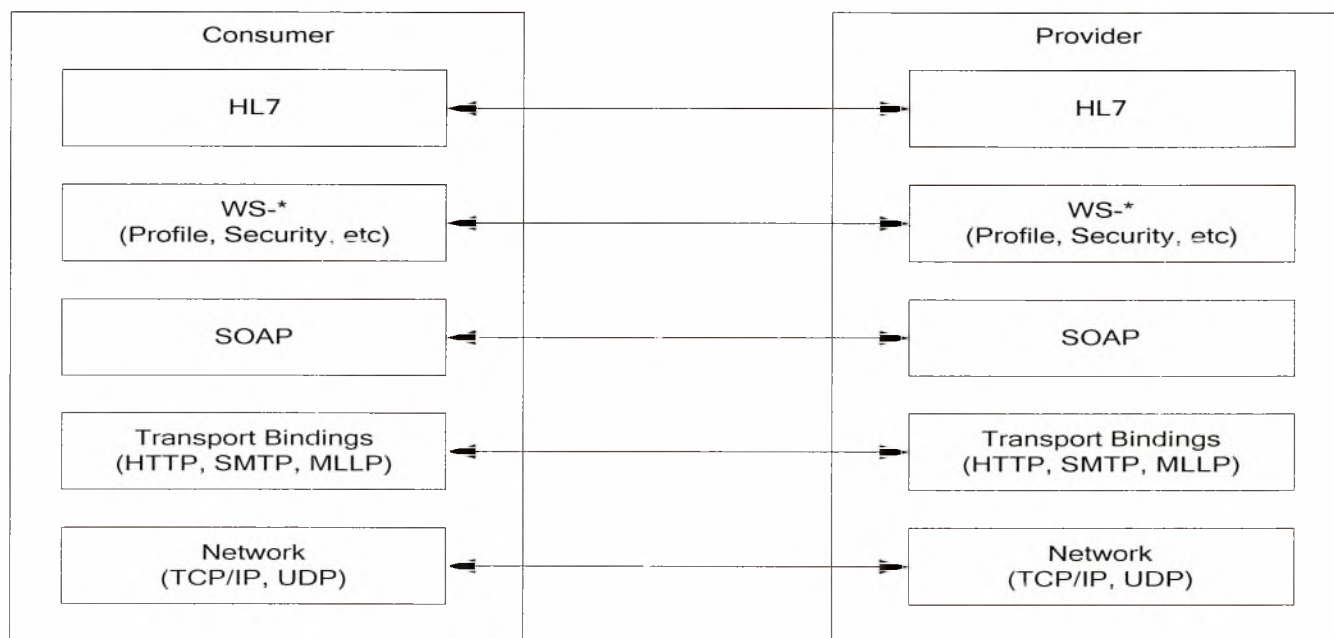
Με βάση λοιπόν, όλα τα παραπάνω, γίνεται κατανοητό ότι μέσω της έρευνας που πραγματοποιήσαμε μελετήσαμε πολλά θέματα και πολλές τεχνολογίες, καθεμία με τα δικά της χαρακτηριστικά και τις δικές τις εφαρμογές. Η έρευνά μας δεν περιορίστηκε μόνο στην Ελλάδα αλλά και σε όλες τις υπόλοιπες αναπτυγμένες χώρες. Ξεκινήσαμε από τη μελέτη των προβλημάτων στην Υγεία, προχωρήσαμε στην ενδελεχή μελέτη των τεχνολογιών που δίνουν λύσεις στα προβλήματα αυτά σε ικανοποιητικό ή σε μη ικανοποιητικό βαθμό, συνεχίσαμε με την εις βάθος έρευνα του προτύπου HL7 και τελειώσαμε με την πειραματική εφαρμογή.

## **8.2. ΜΕΛΛΟΝΤΙΚΕΣ ΕΠΕΚΤΑΣΕΙΣ**

Από τη στιγμή που επιλέξαμε να μετατρέψουμε το HL7 μήνυμα σε XML, καταλαβαίνουμε ότι θα μπορέσουμε να κάνουμε χρήση όλων των Διαδικτυακών Υπηρεσιών (αφού στηρίζονται και χρησιμοποιούν την XML), προκειμένου να μεταφέρουμε HL7 πληροφορίες. Αυτός ο συνδυασμός ονομάζεται HL7 Web Services Profile (WSP). Ένα πιθανό σενάριο επέκτασης της διπλωματικής μας θα μπορούσε να στηριχθεί στη χρήση των Διαδικτυακών Υπηρεσιών και του πρωτοκόλλου SOAP.

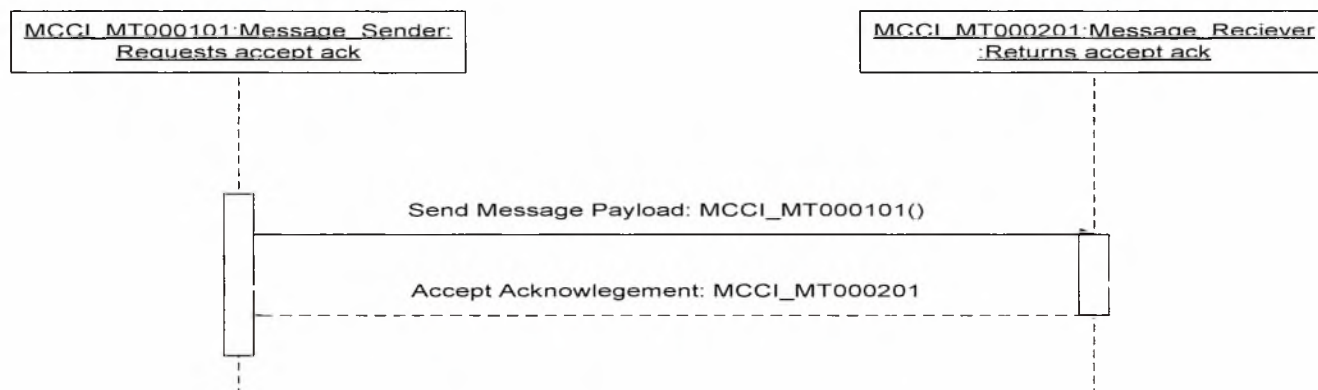
Το ακόλουθο διάγραμμα απεικονίζει τα επίπεδα εφαρμογής που υπάρχουν σε μια επικοινωνία, που βασίζεται στις Διαδικτυακές

Υπηρεσίες και στο πρωτόκολλο SOAP, ανάμεσα σε έναν πελάτη, που επιθυμεί να λάβει ιατρικά δεδομένα, και έναν πάροχο υπηρεσίας, που έχει αναλάβει την υπηρεσία αποστολής των ζητούμενων δεδομένων. Οι δύο συνδιαλλεγόμενοι επιθυμούν να ανταλλάξουν έναν HL7 XML μήνυμα. Στο πάνω επίπεδο βλέπουμε το μήνυμα, κατεβαίνοντας προς τα κάτω έχουμε τα Διαδικτυακά πρωτόκολλα (WS-protocols) που καθορίζουν τις απαιτήσεις των εφαρμογών και ασφάλειας.



Με την εφαρμογή μας, μετατρέπουμε ένα HL7 μήνυμα από την παραδοσιακή μορφή του σε XML. Έτσι θα εισάγουμε το XML πλέον μήνυμά μας σε έναν SOAP φάκελο και θα μεταφερθεί με ασφάλεια (αφού το πρωτόκολλο SOAP, καλύπτει τις προδιαγραφές ασφάλειας, όπως έχουμε αναφέρει στο κεφάλαιο των Διαδικτυακών Υπηρεσιών) μέσω του πρωτοκόλλου HTTP στο διαδίκτυο.

Το ακόλουθο διάγραμμα αλληλεπίδρασης επεξηγεί τη βασική ροή και τους HL7 τύπους μηνυμάτων που θα υπάρχουν στο σενάριο επέκτασης .



Το **Message Receiver** θα είναι η εφαρμογή ή η υπηρεσία που θα παρέχει το WSDL αρχείο και το **Message Sender** θα είναι η εφαρμογή ή η υπηρεσία που θα δρα σαν πελάτης και θα “καταναλώνει” τη Διαδικτυακή Υπηρεσία. Το WSDL αρχείο μπορεί να χωριστεί σε δύο κομμάτια:

Abstract definitions: Θα καθορίζει τους τύπους δεδομένων, τα μηνύματα και τον τρόπο με τον οποίο θα στέλνονται τα μηνύματα με τη χρήση του SOAP.

Concrete definitions: Θα καθορίζει τη φυσική θέση της τελικής εφαρμογής.

Το αρχείο WSDL υπάρχει στο παράρτημα.

Αφού θα διαβαστεί το WSDL αρχείο θα ακολουθεί η κατασκευή του SOAP μηνύματος ως εξής:

- Το Message Sender θα κατασκευάζει την επικεφαλίδα του SOAP μηνύματος.
- Το Message Sender θα κατασκευάζει το body του SOAP μηνύματος, που θα περιέχει τα δεδομένα του HL7 XML μηνύματος.
- Το Message Sender θα ασφαλίζει το SOAP μήνυμα. Η διαδικασία αυτή περιλαμβάνει την κρυπτογράφηση και υπογραφή του μηνύματος.

*Έπειτα θα ακολουθεί η αποστολή του SOAP μηνύματος:*

- Το Message Sender θα στέλνει το SOAP μήνυμα στην επιλεγμένη υπηρεσία.
- Το μήνυμα θα δρομολογείται και θα παραλαμβάνεται από την επιλεγμένη υπηρεσία.

*Μετά την ολοκλήρωση της διαδικασίας αποστολής του SOAP μηνύματος θα λαμβάνει χώρα η διαδικασία παραλαβής του:*

- Το Message Receiver θα δέχεται και θα φιλτράρει το μήνυμα, σύμφωνα με την πολιτική ασφάλειας της Διαδικτυακής Υπηρεσίας, και θα πιστοποιεί την αυθεντικότητά του (ύπαρξη υπογραφής) και την ακεραιότητά του (εφαρμογή κρυπτογράφησης) και τέλος θα το αποσυμπιέζει.

*Στη συνέχεια θα ακολουθεί η διαδικασία κατασκευής της επιβεβαίωσης παραλαβής του SOAP μηνύματος και η αποστολή αυτής:*



- Το Message Receiver θα κατασκευάζει την επιβεβαίωση (acknowledgement) για την παραλαβή του μηνύματος.
- Το Message Receiver θα κατασκευάζει την επικεφαλίδα του SOAP Reply μηνύματος.
- Το Message Receiver θα κατασκευάζει το SOAP Reply body του μηνύματος.
- Το Message Receiver θα ασφαρίζει το SOAP Reply μήνυμα . Τα βήματα αυτής της διαδικασίας είναι: η κρυπτογράφηση και η υπογραφή του μηνύματος.
- Το Message Receiver θα στέλνει το SOAP Reply μήνυμα στο Message Sender.
- Το μήνυμα θα δρομολογείται στο Message Sender.

*Τέλος, θα ακολουθεί η διαδικασία παραλαβής της επιβεβαίωσης του SOAP μηνύματος:*

- Το Message Sender θα λαμβάνει και θα φιλτράρει το Reply μήνυμα και θα πιστοποιεί την αυθεντικότητά του και την ακεραιότητά του.
- Το Message Sender θα εφαρμόζει τις υπόλοιπες προδιαγραφές της Λισδικτυακής Υπηρεσίας και θα αποσυμπιέζει και θα πιστοποιεί την επιβεβαίωση του μηνύματος.

*Τα SOAP request και SOAP reply αρχεία που χρησιμοποιούν τους HL7 XML μετασχηματισμούς, τα παραθέτουμε στο παράρτημα.*

## ΚΕΦΑΛΑΙΟ 9: ΒΙΒΛΙΟΓΡΑΦΙΑ

Οι πηγές που χρησιμοποιήσαμε για την διεκπεραίωση αυτής της διπλωματικής εργασίας είναι οι παρακάτω:

- D.G. Katehakis, S.C. Orphanoudakis, M. Tsiknakis. "*Towards an Integrated Electronic Health Record - Current Status and Challenges*", Business Briefing: Global Healthcare 2002, The Official Publication of the World Medical Association, January 2002.
- S. Orphanoudakis, C. Chronaki, M. Tsiknakis, and S. Kostomanolakis, "*Telematics in Healthcare*", Chapter 10 in Biomedical Image Databases, S. Wong (editor), S. Fletcher, Kluwer Academic Publishers, 1998.
- AIM-1993, "*Research and technology development on telematic systems in healthcare*", DG XIII Adv. Infor. Med., Commission of the European Union, 1993.
- M. Tsiknakis, D. Katehakis, S. Orphanoudakis, "*Information Infrastructure for an Integrated Healthcare Services Network*", IEEE EMBS 3rd International Conference, Information Technology Applications in Biomedicine (ITAB-ITIS 2000), Arlington, Virginia, USA, pp. 278-283, November 9-10, 2000.
- D. Connolly, "*XML: Principles, Tools, and Techniques*", World Wide Web Journal: Volume 2, Issue 4, Fall 1997.
- C. Goldfarb and P. Prescod, "*The XML Handbook*", Upper Saddle River: Prentice Hall, 1998.
- M. Birbek et al., "*Professional XML*", Wrox Press Inc, January 2000.
- G. Coulouris, J. Dollimore, T. Kindberg, "*Distributed Systems: Concepts and Design*", Addison-Wesley Pub Co, 2000, ISBN: 0201619180.

- Elliotte, Rusty Harold, W. Scott Means , “*XML in a Nutshell*”, third edition, 2004.
- Brian Benz, et al, “*XML Programming Bible* “, 2003.
- Michael Fitzgerald, “*Learning XSLT* “, 2004.
- Brian Loesgen, “ *XML Interoperability* “, 2004.
- Shankar R.Ponnekanti and Armando Fox, “ *Interoperability Among Independently Evolving Web Services* “, 2004.
- Jinsoo park and Sudha Ram, “ *Information Systems Interoperability:What Lies Beneath?* “, 2004.
- Robert H. Dolin, Paul V. Biron, “ Using XML as a Supplementary Messaging Syntax“ .

Ένα αρκετά μεγάλο μέρος της έρευνας βασίστηκε σε ιστοσελίδες από το Διαδίκτυο:

- Health Level Seven Official Site: <http://www.hl7.org>
- Online εγκυκλοπαίδεια του SOAP: <http://en.wikipedia.org/wiki/Soap>
- Public library and digital archive, <http://www.ibiblio.org>
- Online web tutorials on XML, XSLT, Xpath: <http://www.w3schools.com>
- Ιστοσελίδα του HL7 στην Ελλάδα: <http://www.hl7hellas.gr>
- Η βιβλιοθήκη του οργανισμού HL7: <http://www.hl7lib.sourceforge.net/>
- Το εγχειρίδιο χρήσης του προγράμματος InterFaceWare: <http://www.interfaceware.com.manual/>
- [http://www.computer-solutions.gr/prod\\_in\\_04.htm](http://www.computer-solutions.gr/prod_in_04.htm)
- HYGEIAnet: <http://www.hygeianet.gr>
- InterMed collaboratory: <http://smi-web.stanford.edu/projects/intermed-web/Overview.html>
- Virtual telemedicine Office: <http://www.telemedical.com>
- The World Wide Web Consortium: <http://www.w3c.org>

- Extensible Markup Language: <http://www.w3c.org/XML>
- Namespaces in XML: <http://www.w3.org/TR/REC-xml-names/>
- OpenLDAP: <http://www.openldap.org>
- OpenMed : <http://sourceforge.net/projects/openmed>

## ΠΑΡΑΡΤΗΜΑ: ΚΩΔΙΚΑΣ

## Παράρτημα Α: Κώδικας των συναρτήσεων της εφαρμογής μας.

Συνάρτηση Get hl7.

[illegible]

```

OR
&nbsp; &nbsp; &nbsp; &nbsp;
<INPUT TYPE="submit" SIZE=12 NAME="submit"
VALUE="Get_the_XML_Format">
</TD></TR>
</TABLE>
</FORM>
</body>
</html>
<?php
    return;
}

```

#### Συνάρτηση Message\_correct

```

Function Message_correct()
{
    global $struct;

?>
<HTML>
<HEAD>
<TITLE>hl72xml </TITLE>
</HEAD>
<BODY BGCOLOR="lightblue" TEXT="black">
<FONT SIZE=+2><B>Check_The_HL7_Format Results</B></FONT><BR>
<HR ALIGN=CENTER WIDTH=90% SIZE=3>
<H2>The HL7 Message format is correct</H2>
</body>
</html>
<?php
    return;
}

```

#### Συνάρτηση hl72xml\_destroy

```

Function hl72xml_destroy($message)
{
?>
<HTML>
<HEAD>
</HEAD>

<BODY BGCOLOR="lightblue" TEXT="black">
<H1>hl72xml - Error</H1>

<?php
    print "<PRE>$message</PRE>\n";
?>
</BODY>
</HTML>
<?php
    exit;
}

```

### Συνάρτηση Process

```

function Process()
{
    // Ανάγνωση του μηνύματος
    $message = $_POST['HL7message'];
    $message = rtrim($message);
    $isFile = 0;
    if (strlen($message) > 0) {
        $Segments = explode("\n", $message);
        if (is_uploaded_file($_FILES['HL7file']['tmp_name'])) {
            unlink($_FILES['HL7file']['tmp_name']);
        }
    } else {
        hl72xml_destroy('THIS IS NOT HL7 MESSAGE');
    }
}

```

```

for ($i = 0; $i < count($Segments); $i++) {
$Segments[$i] = trim($Segments[$i]);
if ($isFile == 0) {
$Segments[$i] = preg_replace("/\\\\\\(\\.)/", "\\1", $Segments[$i]);
}
$SegmentStatus[$i] = '';
}
// Ελέγχει εάν το MSH κομμάτι είναι έγκυρο
if (strlen($Segments[0]) < 20) {
hl7xml_destroy(' The MSH Segment is too short - less than 20
characters');
}
// Ανάγνωση και επεξεργασία του πρώτου segment
$fieldSep = substr($Segments[0], 3, 1);
$compSep = substr($Segments[0], 4, 1);
$repSep = substr($Segments[0], 5, 1);
$escChar = substr($Segments[0], 6, 1);
$subCompSep = substr($Segments[0], 7, 1);
// Διαχωρισμός των Segments
for ($i = 0; $i < count($Segments); $i++) {
$seg = $Segments[$i];
$fields[$i] = array();
$fields[$i] = explode($fieldSep, $seg);
}
// Ελέγχει εάν το 1ο segment είναι το MSH
if ($fields[0][0] != 'MSH') {
hl7xml_destroy('NOT AN HL7 MESSAGE - NO MSH SEGMENT');
}
if (count($fields[0]) < 12) {
hl7xml_destroy('MSH segment too short !');
}
// Ελέγχει την εγκυρότητα των χαρακτήρων
if (strlen($fields[0][1]) < 4) {
$subCompSep = '';
}
if (strlen($fields[0][1]) < 3) {

```



```

$subCompSep = '';
$escChar = '';
}
if (strlen($Fields[0][1]) < 2) {
hl72xml_destroy('MSH.2 is less than 2 characters long');
}
// Ελέγχει την έκδοση του μηνύματος
if (!isset($Fields[0][11])) {
hl72xml_destroy('Missing MSH.12 field [Version ID]');
}
$version = $Fields[0][11];
$version = preg_replace("/\\\$compSep.*"/, '', $version);
if ($version == '2.3.1') {
$XSDdir .= 'v231/xsd/';
} elseif ($version == '2.4') {
$XSDdir .= 'v24/xsd/';
} elseif ($version == '2.5') {
$XSDdir .= 'v25/xsd/';
} else {
hl72xml_destroy('ONLY Version 2.3.1, 2.4 and 2.5 can be checked');
}
// Ελέγχει το Type του μηνύματος
if ((!isset($Fields[0][8])) or (strlen($Fields[0][8]) == 0)) {
hl72xml_destroy('Missing MSH.9 field [Message Type]');
}
$msh9 = explode($compSep, $Fields[0][8]);
if ((isset($msh9[0])) and (strlen($msh9[0]) > 0)) {
$type = $msh9[0];
} else {
hl72xml_destroy('Missing MSH.9.1 component [Message Code]');
}
if ((isset($msh9[1])) and (strlen($msh9[1]) > 0)) {
$event = $msh9[1];
} else {
$ackResp = array ('ACK', 'ADR', 'DOC', 'DSR', 'EAR', 'EDR', 'ERP', 'ESR',
'MCF', 'MFD', 'MFK', 'MFR', 'NMR', 'ORD', 'ORG', 'ORL', 'ORN', 'ORP',

```

```

'ORR', 'ORS', 'OSQ', 'OSR', 'PPT', 'PPV', 'PRR', 'PTR', 'RDY', 'ROR', 'RRA',
'RRD', 'RRE', 'RRG', 'RSP', 'RTB', 'SQR', 'SRR', 'TBR', 'VXR', 'VXX');
if (!in_array($type, $ackResp)) {
hl72xml_destroy('Missing MSH.9.2 component [Trigger Event]');
}
}
if ((isset($msh9[2])) and (strlen($msh9[2]) > 0)) {
$struct = $msh9[2];
} else {
hl72xml_destroy('Missing MSH.9.3 component [Message Structure]');
}
// Μετατροπή σε XML
$XSD = $XSDdir . $struct . '.xsd';
if ($XSDdir != $XSDstructDir) {
if (file_exists($XSDstructDir . $struct . '.xsd')) {
$XSD = $XSDstructDir . $struct . '.xsd';
}
}
ParseXSD($XSD);
$structType = $struct;
if (isset($nameType[$struct])) {
$structType = $nameType[$struct];
if ($nameIs[$structType] != 'COMPLEXTYPE') {
$structType = $struct;
}
}
if ((!isset($Structure[$structType])) or
($Structure[$structType][0] != 'MSH')) {
ParseFail('MSH not defined for messages structure ' . $struct . '[' .
$structType . ']', -1, '');
}
ParseXSD($XSDdir . 'segments.xsd');
ParseXSD($XSDdir . 'fields.xsd');
ParseXSD($XSDdir . 'datatypes.xsd');
$XSD = $XSDdir . 'datatypes.xsd';
if ($_POST['submit'] == 'Get_the_XML_Format') {

```

```

header('Content-type: text/xml');
print '<?xml version="1.0" encoding="UTF-8"?>' . "\n";
print '<' . $struct . ' xmlns="urn:hl7-org:v2xml"
xmlns:xsi="http://www3.org/2001/XMLSchema-instance"
xsi:schemaLocation="urn:hl7-org:v2xml ' . $struct . '.xsd">' . "\n";
} else {
header('Content-type: text/html');
addline(0, 0, "n", "o", "HL7 Message");
}
$thisTree = array();
$lastTree = array();
$tree = '';
for ($segmentNo = 0; $segmentNo < count($Segments); $segmentNo++)
{
if ($SegmentStatus[$segmentNo] == 'unexpected') {
if ($_POST['submit'] == 'Get_the_XML_Format') {
print "<!-- Unexpected Segment ";
}
$fixed = FixXML($Segments[$segmentNo]);
if ($_POST['submit'] == 'Get_the_XML_Format') {
print "\t$fixed -->\n";
} else {
addline(0, 1, "b", "o", "Unexpected Segment - " . $fixed);
}
} elseif ($_POST['submit'] == 'Get_the_XML_Format') {
if ($ParseTree[$segmentNo] == '') {
$thisTree = array();
} else {
$thisTree = explode(' ', $ParseTree[$segmentNo]);
}
$lastDone = count($lastTree) - 1;
$diff = 0;
for($i = 0; $i < count($lastTree); $i++) {
if (($i < count($thisTree)) and ($diff == 0)) {
if ($lastTree[$i] == $thisTree[$i]) {
continue;

```

```

} else {
$diff = 1;
}
}
$XML = $lastTree[$lastDone];
$lastDone--;
if ($XML != "repeat") {
$indent = substr($indent, 0, strlen($indent) - 2);
print $indent;
print " </$XML>\n";
}
}
$diff = 0;
for($i = 0; $i < count($thisTree); $i++) {
if (($i < count($lastTree)) and ($diff == 0)) {
if ($lastTree[$i] == $thisTree[$i]) {
continue;
} else {
$diff = 1;
}
}
$XML = $thisTree[$i];
if ($XML != "repeat") {
print $indent;
print " <$XML>\n";
$indent .= " ";
}
}
$lastTree = $thisTree;
}
$segFields = $Fields[$segmentNo];
$segName = $segFields[0];
$segType = $segName;
if (isset($nameType[$segName])) {
$segType = $nameType[$segName];
if ($nameIs[$segType] != 'COMPLEXTYPE') {

```

```

$segType = $segName;
}
}
if ($segmentNo == 0) {
$segFields[0] = $fieldSep;
$segFields[1] = $compSep . $repSep . $escChar . $subCompSep;
} else {
array_shift($segFields);
}
// Εκτύπωση των segment και των fields
if ($_POST['submit'] == 'Get_the_XML_Format') {
print $indent;
print ' <' . $segName . ">\n";
} else {
$seg = FixEsc($Segments[$segmentNo], '');
addline(0, 1, "n", "c", $Fields[$segmentNo][0] . " - " . $seg);
}
doFields();
if ($_POST['submit'] == 'Get_the_XML_Format') {
print $indent;
print ' </' . $segName . ">\n";
}
}
if ($_POST['submit'] == 'Get_the_XML_Format') {
if ($indent != '') {
$i = count($lastTree) - 1;
while ($indent != '') {
$XML = $lastTree[$i];
if ($XML != "repeat") {
$indent = substr($indent, 0, strlen($indent) - 2);
print $indent;
print " </$XML>\n";
$lastXML = $XML;
}
$i--;
}
}

```

```

}
print '</' . $struct . ">\n";
} else {
$title = " HL7 View";
$titles[0] = "<B><U> HL7 View</U></B>";
mktframes($title, $titles);
}
}

```

Παράρτημα Β: Κώδικας του WSDL αρχείου.

```

1 <definitions xmlns="http://schemas.xmlsoap.org/wsdl/"
2   xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
3   xmlns:http="http://schemas.xmlsoap.org/wsdl/http/"
4   xmlns:xs=http://www.w3.org/2001/XMLSchema
5   xmlns:soapenc=http://schemas.xmlsoap.org/soap/encoding/
6   xmlns:mime="http://schemas.xmlsoap.org/wsdl/mime/" xmlns:ns="u
7   rn:hl7-org:v3:ws"
8   xmlns:hl7="urn: hl7-org: v3"   targetNamespace="urn: hl7-
9   org:v3:ws">
10  <types>
11    <xs: schema elementFormDefault="qualified"
12      targetNamespace="urn: hl7-org: v3">
13      <!-- Include the schema for the messages itself -->
14      <xs: include schemaLocation="POLB_MT002102.xsd"/>
15      <!-- Include the schema for the messages wrapper -->
16      <xs: include schemaLocation="MCCI_MT000101.xsd"/>
17      <!-- Include the schema for the acknowledge message -->
18      <xs: include schemaLocation="MCCI_MT000201.xsd"/>
19      <!-- Define the body of the message -->
20      <xs: complexType name="MCCI_MT000101.ControlActEvent">
21        <xs: sequence>
22          <xs: element name="observationOrder"
23            type="hl7:POLB_MT002102"/>

```

```

20</xs: sequence>
21</xs: complexType>
22<!-- Define the element for the message -->
23<xs: element name="elPOLB_MT002102"
type="hl7:MCCI_MT000101.Message"/>
24<!-- Define the element for the acknowledge -->
25<xs: element name="elMCCI_MT000201"
type="hl7:MCCI_MT000201.Message"/>
26</xs: schema>
27</types>
28<message name="msgPOLB_MT002102In">
29<part name="ptPOLB_MT002102"
element="hl7:elPOLB_MT002102"/>
30</message>
31<message name="msgPOLB_MT002102Out">
32<part name="ptMCCI_MT000201"
element="hl7:elMCCI_MT000201"/>
33</message>
34<portType name="portPOLB_MT002102">
35<operation name="opPOLB_MT002102">
36  <input message="ns: msgPOLB_MT002102In"/>
37<output message="ns: msgPOLB_MT002102Out"/>
38</operation>
39</portType>
40<binding name="bindingPOLB_MT002102" type="ns:
portPOLB_MT002102">
41<soap:binding style="document"
transport="http://schemas.xmlsoap.org/soap/http"/>
42<operation name="opPOLB_MT002102">
43<soap: operation soapAction="urn:#opPOLB_MT002102"
style="document"/>
44<input>
45<soap: body use="literal"/>

```

```

46</input>
47<output>
48<soap: body use="literal"/>
49</output>
50</operation>
51</binding>
52<service name="svcPOLB_MT002102">
53<port name="POLB_MT002102" binding="ns:
bindingPOLB_MT002102">
54  <soap: address location="No Target Address"/>
55</port>
56</service>
57</definitions>

```

Συνοπτικά αναφέρουμε κάποια σχόλια για τις γραμμές του κώδικα:

Γραμμές 1-7: Καθορισμός των namespaces.

Γραμμές 8-27: Δήλωση των τύπων θα χρησιμοποιηθούν για τον καθορισμό των μηνυμάτων.

Γραμμή 11: Το αρχείο schema που καθορίζει τη συγκεκριμένη HL7 έκδοση 3 του μηνύματος. Μπορούν να συμπεριληφθούν και διαφορετικά schema αρχεία για τον καθορισμό άλλων HL7 μηνυμάτων της έκδοσης 3.

Γραμμή 13: Το schema αρχείο που δείχνει την τοποθεσία του μηνύματος.

Γραμμή 15: Το schema αρχείο που δείχνει που υπάρχει η επιβεβαίωση του μηνύματος.

Γραμμές 17-21: Εδώ ο τύπος για το ControlActEvent επαναπροσδιορίζεται για να προσδιορίσει σωστά το **payload** του μηνύματος HL7 της έκδοσης 3, ώστε να είναι τύπου POLB\_MT002102.

Γραμμή 23: Το στοιχείο που καθορίζει τον τύπο της παραμέτρου εισόδου.



Γραμμή 25: Το στοιχείο που καθορίζει τον τύπο της παραμέτρου εξόδου.

Γραμμές 28-30: Εδώ καθορίζεται η παράμετρος εισόδου. Η σύμβαση που χρησιμοποιείται για να καθορίσει τις παραμέτρους εισόδου είναι: “msg<HL7 Message Name>In”, σε αυτήν την περίπτωση είναι msgPOLB\_MT002102In. Η σύμβαση είναι αυθαίρετη και μπορεί να αλλάξει.

Γραμμές 31-33: Εδώ καθορίζεται η παράμετρος εξόδου. Η σύμβαση που χρησιμοποιείται είναι: “msg<HL7 Message Name>Out”, σε αυτήν την περίπτωση sgPOLB\_MT002102Out. Η σύμβαση είναι αυθαίρετη και μπορεί να αλλάξει.

Γραμμές 34-39: Εδώ οι παράμετροι εισόδου και εξόδου συνδυάζονται για να καθορίσουν το port type. Η σύμβαση που χρησιμοποιείται είναι: “port<HL7 Message Name>”, σε αυτήν την περίπτωση portPOLB\_MT002102. Η σύμβαση είναι αυθαίρετη και μπορεί να αλλάξει.

Γραμμές 40-51: Το port type που καθορίστηκε πριν συνδέεται σε συγκεκριμένο πρωτόκολλο, εδώ στο HTTP. Η σύμβαση που χρησιμοποιείται για να καθορίσει τη σύνδεση είναι: “binding<HL7 Message Name>”, σε αυτήν την περίπτωση bindingPOLB\_MT002102. Η σύμβαση είναι αυθαίρετη και μπορεί να αλλάξει.

Γραμμή 41: Το πρωτόκολλο που επιλέγεται εδώ, όπως είπαμε είναι το HTTP. Η γραμμή αυτή καθορίζει το στυλ που θα χρησιμοποιηθεί για την εισαγωγή του HL7 μηνύματος στο SOAP body.

Γραμμή 45: Αυτή η ρύθμιση καθορίζει την κωδικοποίηση του HL7 μηνύματος, ώστε να συμπεριληφθεί στο SOAP body. Η ρύθμιση “literal” διευκρινίζει ότι το μήνυμα πρέπει να συμπεριληφθεί αυτούσιο μέσα στο SOAP body.

Γραμμή 46: Αυτή η ρύθμιση καθορίζει την κωδικοποίηση του HL7 μηνύματος, που θα συμπεριληφθεί στο SOAP body, για την παραγωγή της παραμέτρου εξόδου.

Γραμμές 52-56: Σε αυτό το τμήμα ο τύπος της Διαδικτυακής Υπηρεσίας που δηλώνεται στο έγγραφο WSDL συνδέεται σε μια πραγματική εφαρμογή. Η σύμβαση που χρησιμοποιείται είναι: “svc<HL7 Message Name>”, σε αυτήν την περίπτωση svcPOLB\_MT002102.

Παράρτημα Γ: Κώδικας του SOAP reply και του SOAP request μηνύματος.

### **SOAP Request**

POST /targetAddress/svcPOLB\_MT002102.asmx HTTP/1.1

Host: hostAddress

Content-Type: text/xml; charset=utf-8

Content-Length: length

SOAPAction: "urn:#opPOLB\_MT002102"

<?xml version="1.0" encoding="utf-8"?>

<soap:Envelope xmlns:xsi=<http://www.w3.org/2001/XMLSchema-instance>

xmlns:xsd=<http://www.w3.org/2001/XMLSchema>

xmlns:soap=<http://schemas.xmlsoap.org/soap/envelope/>

xmlns:hl7="urn:hl7-org:v3">

<soap:Body>

<elPOLB\_MT002102 xmlns="urn:hl7-org:v3"

```

        xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance"
        xsi:schemaLocation="urn:hl7-
org:v3.\schemas\envPOLB_MT002102.xsd">
        <hl7:id root='2.16.840.1.113883.9876.349'
extension='347782'/>
        <hl7:creationTime value='20030103170600'/>
        <hl7:sender>
        <hl7:servedBy>
        <hl7:id root='2.16.840.1.113883.9876.349'
extension='EpicCare'/>
        <hl7:name>EpicCare Patient Medical
Records</hl7:name>
        </hl7:servedBy>
        </hl7:sender>
        <hl7:receiver>
        <hl7:servedBy>
        <hl7:id root='2.16.840.1.113883.9876.369'
extension='InternalLab'/>
        <hl7:name>NeoTool Excellent Lab
Systems</hl7:name>
        </hl7:servedBy>
        </hl7:receiver>
        <hl7:versionId>v3r1b3</hl7:versionId>
        <hl7:interactionId root='2.16.840.1.113883.9876.5'
extension='POLB_IN002120'/>
        <hl7:processingCode code='P'/>
        <hl7:processingModeCode code='T'/>
        <hl7:acceptAckCode code='AL'/>
        <hl7:applicationAckCode code='AL'/>
        <hl7:hasPayload>
        <hl7:observationOrder>

```

```

    <hl7:id root='2.16.840.1.113883.9876.349'
extension='34522'/>
    <hl7:code code='P3-30100'
      codeSystem='2.16.840.1.113883.6.5'
      codeSystemName='SNOMED'
      codeSystemVersion='20020829'
      displayName='Complete _u66 Flood Count'/>
    <hl7:effectiveTime value='20030103170100'/>
    <hl7:priorityCode code='N' displayName='Normal'/>
    <hl7:participant typeCode='VRF'>
      <hl7:assignedEntity>
        <hl7:id root='2.16.840.1.113883.9876.210.3'
extension='5332443'/>
        <hl7:assigneePerson>
          <hl7:name>
            <hl7:given>Keiko</hl7:given>
            <hl7:family>Jones</hl7:family>
          </hl7:name>
        </hl7:assigneePerson>
      </hl7:assignedEntity>
    </hl7:participant>
    <hl7:patientParticipation>
      <hl7:patient>
        <hl7:id root='2.16.840.1.113883.9876.211'
extension='344253425'/>
        <hl7:addr>
          <hl7:streetAddressLine>2222 Home
Street</hl7:streetAddressLine>
          <hl7:city>Ann Arbor</hl7:city>
          <hl7:state>MI</hl7:state>
          <hl7:postalCode>99999</hl7:postalCode>
          <hl7:country>USA</hl7:country>
        </hl7:addr>
      </hl7:patient>
    </hl7:patientParticipation>
  </hl7:encounter>

```

```

        <hl7:telecom value='tel:213-555-4344'/>
        <hl7:patientPerson>
            <hl7:id root='2.16.840.1.113883.4.1'
extension='333224444'/>
            <hl7:name>
                <hl7:given>George</hl7:given>
                <hl7:given>Simon</hl7:given>
                <hl7:family>Wigny</hl7:family>
            </hl7:name>
            <hl7:administrativeGenderCode code='M'
codeSystem='2.16.840.1.113883.5.1'/>
            <hl7:birthTime value='19740423'/>
        </hl7:patientPerson>
    </hl7:patient>
</hl7:patientParticipation>
</hl7:observationOrder>
</hl7:hasPayload>
</elPOLB_MT002102>
</soap:Body>
</soap:Envelope>

```

## SOAP Reply

HTTP/1.1 200 OK

Content-Type: text/xml; charset=utf-8

Content-Length: length

```

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
    <soap:Body>

```

```

<elMCCI_MT000201 xmlns="urn:hl7-org:v3"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance"
    xsi:schemaLocation="urn:hl7-
org:v3./schemas/MCCI_MT000201.xsd">
    <hl7:id root='2.16.840.1.113883.9876.369'
extension='231112'/>
    <hl7:creationTime value='2002106170601'/>
    <hl7:sender>
        <hl7:servedBy>
            <hl7:id root='2.16.840.1.113883.9876.369'
extension='Internal_Lab'/>
            <hl7:name>NeoTool Excellent Lab
Systems</hl7:name>
        </hl7:servedBy>
    </hl7:sender>
    <hl7:receiver>
        <hl7:servedBy>
            <hl7:id root='2.16.840.1.113883.9876.349'
extension='EpicCare'/>
            <hl7:name>EpicCare Patient Medical
Record</hl7:name>
        </hl7:servedBy>
    </hl7:receiver>
    <hl7:versionId>v3r1b3</hl7:versionId>
    <hl7:interactionId cf5 root='2.16.840.1.113883.9876.5'
        extension='MCCI_IN000200'/>
    <hl7:processingCode code='P'/>
    <hl7:processingMode code='T'/>
    <hl7:has typeCode='AR'>
        <hl7:errorDetailCode code='42'/>
        <hl7:noteText>This is a note explaining code
42</hl7:noteText>

```

```
<hl7:acknowledges>
  <hl7:id root='2.16.840.1.113883.9876.349'
extension='347782' />
</hl7:acknowledges>
</hl7:has>
</eIMCCI_MT000201>
</soap:Body>
</soap:Envelope>
```



ΠΑΝΕΠΙΣΤΗΜΙΟ  
ΘΕΣΣΑΛΙΑΣ



004000085993